

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIF 93940

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

CLASSIFICATION TECHNIQUES FOR
MULTIVARIATE DATA ANALYSIS

by

Jin Ki Lee

March 1980

Thesis Advisor:

F. R. Richards

Approved for public release; distribution unlimited.

T195235

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Classification Techniques for Multivariate Data Analysis		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis March 1980
7. AUTHOR(s) Jin Ki Lee		6. PERFORMING ORG. REPORT NUMBER
8. CONTRACT OR GRANT NUMBER(s)		
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE March 28, 1980
		13. NUMBER OF PAGES 120
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) multivariate analysis, principal components analysis, variance-covariance matrix, correlation, lagrangian technique discriminant analysis, distance measure (metric), hierarchical clustering, nonhierarchical clustering, similarity matrix		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The multivariate analysis techniques of cluster analysis, principal components analysis, and discriminant analysis are exam- ined in this thesis. The theory and applications of each of the techniques are discussed. Computer software available at the Naval Postgraduate School is discussed and sample jobs are included. A hierarchical cluster analysis algorithm, available in the IMSL software package, is applied to a set of data extracted from a		

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

group of subjects for the purpose of partitioning a collection of 26 attributes of a weapon system into six clusters of super-attributes.

A nonhierarchical clustering procedure, principal components analysis, and discriminant analysis were all applied to a collection of data on tanks considering of twenty-four observations of ten attributes of tanks. The cluster analysis shows that the tanks cluster somewhat naturally by nationality. The principal components analysis and the discriminant analysis show that tank weight is the single most important discriminator among nationality.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Approved for public release, distribution unlimited

Classification Techniques for
Multivariate Data Analysis

by

Jin Ki Lee
Lieutenant Colonel, Korean Army
B.S., Korean Military Academy, 1966

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1980

ABSTRACT

The multivariate analysis techniques of cluster analysis, principal components analysis, and discriminant analysis are examined in this thesis. The theory and applications of each of the techniques are discussed. Computer software available at the Naval Postgraduate School is discussed and sample jobs are included.

A hierarchical cluster analysis algorithm, available in the IMSL software package, is applied to a set of data extracted from a group of subjects for the purpose of partitioning a collection of 26 attributes of a weapon system into six clusters of superattributes.

A nonhierarchical clustering procedure, principal components analysis, and discriminant analysis were all applied to a collection of data on tanks considering of twenty-four observations of ten attributes of tanks. The cluster analysis shows that the tanks cluster somewhat naturally by nationality. The principal components analysis and the discriminant analysis show that tank weight is the single most important discriminator among nationality.

TABLE OF CONTENTS

	PAGE
I. DISCUSSION OF MULTIVARIATE DATA ANALYSIS . . .	8
II. PRINCIPAL COMPONENTS ANALYSIS	12
III. DISCRIMINANT ANALYSIS	19
A. Introduction	19
B. Theory	20
IV. CLUSTER ANALYSIS	32
A. Origin and Theory	32
B. Measure of Distance	34
C. Hierarchical Clustering	41
D. Nonhierarchical Clustering	47
V. ANALYSIS OF MULTIVARIATE UTILITY DATA . . .	52
VI. ANALYSIS OF ARMY TANK DATA	62
A. Data Structure	62
B. Nonhierarchical Cluster Analysis of Tank Data	64
C. Principal Components Analysis	69
D. Discriminant Analysis	73
VII. CONCLUSION	80
APPENDIX A: Program List for Hierarchical Clustering	81
APPENDIX B: Program List for MIKCA	88
APPENDIX C: Program List for Principal Component Analysis	116
APPENDIX D: Program List for Discriminant Analysis .	117
BIBLIOGRAPHY	118
INITIAL DISTRIBUTION LIST	120

LIST OF TABLES

	PAGE
I. ILLUSTRATIVE DATA MATRIX	9
II. DATA MATRIX	55
III. SIMILARITY MATRIX FOR SUPERATTRIBUTE DETERMINATION	58
IV. SUPERATTRIBUTES	61
V. THE FINAL CLUSTER SOLUTIONS	68

LIST OF FIGURES

	PAGE
1. EUCLIDEAN DISTANCE	38
2. TREE FOR HIERARCHICAL CLUSTERING	42
3. LOWER TRIANGULAR SIMILARITY MATRIX	43
4. TREE FOR 26 ATTRIBUTES	59
5. MIKCA FLOW CHART	65
6. SUMMARY TABLE OF PRINCIPAL COMPONENTS ANALYSIS ON TANK	70
7. GRAPHICAL PRESENTATION	71
8. FACTOR SCORE COEFFICIENTS	72
9. SUMMARY TABLE OF DISCRIMINANT ANALYSIS ON TANK .	75
10. CANONICAL DISCRIMINANT FUNCTION COEFFICIENTS. .	76
11. SCATTER PLOT FOR ALL GROUPS	78

I. DISCUSSION OF MULTIVARIATE DATA ANALYSIS

A. INTRODUCTION

As a set of statistical techniques, multivariate data analysis is concerned with data collected on several dimensions of the same observations. Techniques can be used for many purpose in the behavioral, mathematical, and administrative sciences - ranging from rigidly controlled experiments to explain relationships assumed to be present in a large mass of data to attempts to cluster similar elements or to find functions of the variables that will best discriminate among preselected subpopulations of the observations.

The heart of any multivariate analysis consists of the data matrix. This matrix is a table that gives the results of a number of observations on a number of variables simultaneously (Table I).

Illustrative Data Matrix

Observations	Variables				
	1	2	3	 j	 p
1	x_{11}	x_{12}	x_{13}	x_{1j}	x_{1p}
2	x_{21}	x_{22}	x_{23}	x_{2j}	x_{2p}
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
i	x_{i1}	x_{i2}	x_{i3}	x_{ij}	x_{ip}
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
n	x_{n1}	x_{n2}	x_{n3}	x_{nj}	x_{np}

TABLE I.

The table consists of a set of observations (the n rows) and a set of measurements on those observations (the p columns). Cell entries represent the value x_{ij} of observation i on variable j . The values are characteristics of the observations and serve to define the observations in any specific study. The cell values may consist of nominal, ordinal, interval, or ratio-scaled measurements, or various combinations of these across columns.

In a general sense "multivariate" analysis would concern two main features:

1. The multivariate character lies in the multiplicity of the p variables, not in the size of the set n .
2. The variables are dependent among themselves so that we can not split off one or more from the others and consider it by itself. The variables must be considered together.

There are three characteristics often used as a basis for the classification of multivariate analysis:

1. whether one's principal focus is on the objects or on the variables of the data matrix;
2. whether the data matrix is partitioned into criterion and independent subsets, and the number of variables in each;
3. whether the cell values represent nominal, ordinal, or interval scale measurements.

This classification results in four major subdivisions of interest:

1. single criterion, multiple predictor association, including multiple regression, analysis of variance and covariance, and two-group discriminant analysis;

2. multiple criterion, multiple predictor association, including canonical correlation, multivariate analysis of variance and covariance, and multiple discriminant analysis;
3. analysis of variable interdependence, including factor analysis, multidimensional scaling, and other types of dimension-reducing methods;
4. analysis of interobject similarity, including cluster analysis and other types of grouping procedures.

The first two categories involve dependence structures where the data matrix is partitioned into criterion and independent subsets; in both cases interest is focused on the variables. The last two categories are concerned with interdependence - either focusing on variables or on observations. Within each of four categories, various techniques are differentiated in terms of the type of scale assumed.

In this research, we consider only the following techniques of multivariate analysis:

1. Principal components analysis
2. Discriminant analysis
3. Cluster analysis

II. PRINCIPAL COMPONENTS ANALYSIS

The basic idea of principal components analysis is to describe the dispersion of an array of n points in p -dimensional space by introducing a new set of orthogonal linear coordinates so that the sample variances of the given data points with respect to these derived coordinates are in decreasing order of magnitude. Thus the first principal component is such that the projection of given points onto it have maximum variance among all possible linear coordinates; the second principal component has maximum variance subject to being orthogonal to the first; and so on.

Suppose that the random variables $X_1, X_2, \dots, X_p$ of interest have a certain multivariate distribution with finite mean vector u and variance-covariance matrix Σ .

From this population a sample of n independent observation vectors has been drawn. The observation can be written as the usual $n \times p$ data matrix.

$$X = \begin{bmatrix} x_{11} & \dots & x_{1p} \\ \vdots & & \vdots \\ x_{n1} & \dots & x_{np} \end{bmatrix} = \begin{bmatrix} X'_1 \\ \vdots \\ X'_n \end{bmatrix} \quad (1)$$

The estimate of Σ will be the usual sample variance-covariance matrix S defined as follows:

$$S = \frac{1}{n-1} A$$

$$A = \sum_{j=1}^n (X_j - \bar{X})(X_j - \bar{X})' \quad (2)$$

The information we shall need for our principal components analysis will be contained in S . However, it will be necessary to make a choice of measures of dependence: should we work with the variances and covariances of the observations, and carry out the analysis in original unit of the responses, or would a more accurate picture of the dependence pattern be obtained if each x_{ij} were transformed to a standardized score

$$z_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}$$

and the correlation matrix R employed? The components obtained from S and R in general not the same, nor is it possible to pass from one solution to the other by a simple scaling of the coefficients.

If the responses are in widely different units (i.e., number of crew, weight in tons, speed in kilometer per hour, etc.) with large differences in the magnitudes, linear compounds of original quantities would have little

meaning and standardized variates and correlation matrix should be employed. Conversely, if the responses are reasonably commensurable, the covariance form has a greater statistical appeal, for the i -th principal component is that linear compound of the responses which explains the i -th largest portion of the total response variance, and maximization of such total variance of standard scores is rather artificial.

The first principal component of the complex of sample values of the responses $X_1, X_2, \dots, X_p$ is the linear compound

$$Y_1 = a_{11}X_1 + \dots + a_{p1}X_p \quad (3)$$

whose coefficients a_{i1} are the elements of the eigenvector associated with the greatest eigenvalue λ_1 of the sample variance-covariance matrix of the responses. The a_{i1} are unique up to multiplication by a scale factor, and if they are scaled so that $a'_1 a_1 = 1$, the eigenvalue λ_1 is interpretable as the sample variance of Y_1 .

Numerical representation of the first principal component is to find the vector A_1 such that

$$\begin{aligned} Y_1 &= a_{11}X_1 + \dots + a_{p1}X_p \\ &= A'_1 X \end{aligned} \quad (4)$$

which maximizes sample variance

$$\begin{aligned}
 s_{Y_1}^2 &= \sum_{i=1}^p \sum_{j=1}^p a_{i1} a_{j1} s_{ij} \\
 &= A_1' S A_1
 \end{aligned} \tag{5}$$

for all coefficient vectors normalized so that $A_1' A_1 = 1$. To determine the coefficients, the normalization constraint is introduced by means of Lagrange multiplier and the resulting expression is differentiated with respect to A_1 :

$$\begin{aligned}
 \frac{\partial}{\partial A_1} [s_{Y_1}^2 - \lambda_1 (1 - A_1' A_1)] &= \frac{\partial}{\partial A_1} [A_1' S A_1 + \lambda_1 (1 - A_1' A_1)] \\
 &= 2(S - \lambda_1 I) A_1
 \end{aligned} \tag{6}$$

The coefficients must satisfy the p simultaneous linear equations.

$$(S - \lambda_1 I) A_1 = 0 \tag{7}$$

If the solution to these equation is to be other than the null vector, the value of λ_1 must be chosen so that

$$|S - \lambda_1 I| = 0 \tag{8}$$

λ_1 is thus an eigenvalue of the variance-covariance matrix, and A_1 is its associated eigenvector. To determine which of the p eigenvalues should be used, premultiply the

the system of equation (7) by A_1' . Since $A_1' A_1 = 1$, it follows that

$$\lambda_1 = A_1' S A_1 = s_{Y_1}^2$$

But the coefficient vecotr was chosen to maximize this variance, and therefore, λ_1 must be the greatest eigenvalue of S .

The second principal component is that linear compound

$$Y_2 = a_{12}X_1 + \dots\dots\dots + a_{p2} X_p \quad (9)$$

whose coefficients have been chosen, subject to the constraints

$$\begin{aligned} A_2' A_2 &= 1 \\ A_1' A_2 &= 0 \end{aligned} \quad (10)$$

so that the variance of Y_2 , $A_2' S A_2$, is a maximum. The first constraint is merely a scaling to assure the uniqueness of the coefficients, while the second requires that A_1 and A_2 be orthogonal.

The coefficients of the second component can also be found by the Lagrangian technique with two multipliers λ_2 and μ . Differentiating this with respect to A_2 gives:

$$\begin{aligned} & \frac{\partial}{\partial A_2} [A_2' S A_2 + \lambda_2 (1 - A_2' A_2) + \mu A_1' A_2] \\ & = 2(S - \lambda_2 I)A_2 + \mu A_1 \end{aligned} \quad (11)$$

If the right-hand side is set equal to 0 and premultiplied by A_1' , it follows from the normalization and orthogonality conditions that

$$2 A_2' S A_2 + \mu = 0 \quad (12)$$

Similar premultiplication of the equation (7) by A_2' implies that

$$A_2' S A_2 = 0 \quad (13)$$

and hence $\mu = 0$. The second vector must satisfy

$$(S - \lambda_2 I)A_2 = 0 \quad (14)$$

And it follows that the coefficients of the second component are thus the elements of the eigenvector corresponding to the second greatest eigenvalue. The remaining principal components are found in their turn in the same manner from the other eigenvectors.

Thus the j -th principal component of the sample of p -variate observations is the linear compound

$$Y_j = a_{1j}X_1 + \dots + a_{pj}X_p \quad (15)$$

whose coefficients are the elements of the eigenvector of the sample variance-covariance matrix S corresponding to the j -th largest eigenvalue λ_j . If $\lambda_i \neq \lambda_j$, the coefficients of the i -th and j -th components are necessarily orthogonal; if $\lambda_i = \lambda_j$, the elements can be chosen to be orthogonal, although an infinity of such orthogonal vectors exists. The sample variance of the j -th components is λ_j , and the total system variance is thus

$$\lambda_1 + \lambda_2 + \dots + \lambda_p = \text{tr } S \quad (16)$$

The importance of the j -th component in a more parsimonious description of the system is measured by

$$\frac{\lambda_j}{\text{tr } S} \quad (17)$$

which gives the fraction of the total variance contributed to the j -th component.

III. DISCRIMINANT ANALYSIS

A. INTRODUCTION

The basic idea of discriminant analysis consists of assigning an individual from a group of individuals to one of several known or unknown distinct populations, on the basis of observations on several characters of the individual or group and a sample of observations on these characters from the populations if these are unknown.

Fisher (1936) was the first to suggest a linear function of variables representing different characters, hereafter called the linear discriminant function (discriminator) for classifying an individual into one of two populations. Later research extended the analysis to classification into one of k populations.

For the univariate case Fisher suggested a rule which classifies an observation x into the i -th univariate population if

$$x - \bar{x}_i = \min (x - \bar{x}_1, x - \bar{x}_2), \quad i = 1, 2 \quad (18)$$

where $\bar{x}_i$ is the sample mean based on a sample of size N_i from i -th population. For two p -variate populations π_1 and π_2 (with the same covariance matrix) Fisher replaced the vector random variable by an optimum linear combination of its components obtained by maximizing the

ratio of the difference of the expected values of a linear combination under π_1 and π_2 to its standard deviation. He then used his univariate discrimination method with this optimum linear combination of components as the random variable.

Rao (1948) considered the problem of classifying people into one of these populations castes of India. He assumed that each of the three populations could be characterized by four variables - structure (x_1), sitting height (x_2), nasal depth (x_3), and nasal height (x_4) - of each member of the population. On the basis of sample observations on these characters from the three populations the problem is to classify an individual with observation $X = (x_1, x_2, x_3, x_4)^T$ into one of three populations. He used a linear discriminator to obtain the solution.

B. THEORY

In general, the underlying assumptions of discriminant analysis are:

1. the groups being investigated are discrete and identifiable;
2. each observation in each group can be described by a set of measurements on p characteristics or variables;
3. these p variables are assumed to have a multivariate normal distribution in each population.

The purposes of discriminant analysis are:

1. to test for mean group differences and to describe the overlaps among groups;
2. to construct classification schemes based upon the set of p variables in order to assign previously unclassified observations to the appropriate groups.

Hence, the problem of studying the direction of group differences is, equivalently, a problem of finding a linear combination of the original independent variables that shows large differences in group means. In short, discriminant analysis is a method for determining such linear combinations.

The first step toward determining a linear combination of a set of variables such that several group means on this linear combination will differ widely among themselves, is to decide on a criterion for measuring such group-mean differences. Once a linear combination has been constructed, that means there is just a single transformed variable. Hence, the F -ratio for testing the significance of the over all difference among several group means on a single variable suggests an appropriate criterion.

$$F = \frac{v' B v}{v' W v} = \lambda \quad (19)$$

where $v' = (v_1, v_2, \dots, v_p)$, a set of weight which maximizes λ .

$$B = \sum_{i=1}^G N_i (\bar{x}_{i.} - \bar{x}_{..}) (\bar{x}_{i.} - \bar{x}_{..})'$$

$$W = \sum_{i=1}^G \sum_{j=1}^{n_i} (x_{ij} - \bar{x}_{i.}) (x_{ij} - \bar{x}_{i.})'$$

x_{ij} is the j th observation vector in the i -th group.

$\bar{x}_{..}$ is the grand mean vector of the data.

G is the number of groups.

n_i is the number of observations in the i th group.

Prime notation indicates transpose.

This ratio λ , called the discriminant criterion, was originally proposed by Fisher in connection with his two-group discriminant function. Once a criterion for group differentiation has been determined, a set of weights, $(v_1, v_2, \dots, v_p)$, which maximizes this criterion, should be determined. This is accomplished by taking the partial derivative of λ with respect to each component v_i of v and setting the result equal to zero.

$$\begin{aligned} \frac{\partial \lambda}{\partial v} &= \frac{2[(Bv)(v'Wv) - (v'Wv)(Wv)]}{(v'Wv)^2} \\ &= \frac{2(Bv - Wv)}{v'Wv} = 0 \end{aligned} \tag{20}$$

which is equivalent to

$$\begin{aligned}(B - \lambda W)v &= 0 \\ (W^{-1}B - \lambda I)v &= 0\end{aligned}\tag{21}$$

This equation is of the form

$$(A - \lambda I)v = 0\tag{22}$$

It's solution, yielding the eigenvalues λ_p and associated eigenvectors V_p of the matrix A , is therefore the same as in the principal components analysis, and thus the solved problem satisfies the problem of maximizing the discriminant criterion.

In the last equation, the number of non-zero eigenvalues of a square matrix A is equal to the rank of A . With $W^{-1}B$ playing the role of A , the number of non-zero eigenvalues depends on the rank of B , since the rank of the product of two matrices can not exceed the smaller of the two factor matrices' ranks, and W^{-1} (being nonsingular) must be of full rank p , while the rank of B is usually smaller than p . Thus it is possible to denote the rank of B by $r = \min (G-1, p)$.

From the fact that the eigenvalues λ_p are the values assumed by the discriminant criterion for linear combination using the elements of the corresponding eigenvectors P as combining weights, it is clear that the eigenvector

$V_1' = (v_{11}, v_{12}, \dots, v_{1p})$ provides a set of weights such that the transformed variable

$$Y_1 = v_{11}X_1 + v_{12}X_2 + \dots + v_{1p}X_p \quad (23)$$

has the largest discriminant-criterion, λ , achievable by any linear combination of the p independent variables.

What are the properties of the remaining eigenvectors, $v_2, v_3, \dots, v_p$? The second discriminant function $Y_2 = v_{21}X_1 + v_{22}X_2 + \dots + v_{2p}X_p$ whose weights are the elements of the eigenvector v_2 associated with the second largest eigenvalue λ_2 of $W^{-1}B$ has the largest discriminant-criterion among those linear combinations of the X_i that are uncorrelated with the first discriminant function in the total sample observation. Its proof is analogous of that of principal components analysis. Each discriminant function has a relative (or conditional) maximum value for its discriminant criterion. Therefore, it needs nonly to show that Y_2 is uncorrelated with Y_1 . Noting that this correlation is proportional to $v_1'Tv_2$ (where $T = W + B$), we have to prove that $v_1'Tv_2 = 0$.

$$(B - \lambda_i W)v_i = 0 \quad \text{for each } i \quad (24)$$

hence,

$$Bv_1 = \lambda_1 Wv_1 \quad \text{and} \quad Bv_2 = \lambda_2 Wv_2$$

premultiplying these equations by v_2' and v_1' respectively,

$$\begin{aligned} v_2' B v_1 &= \lambda_1 v_2' W v_1 \\ v_1' B v_2 &= \lambda_2 v_1' W v_2 \end{aligned} \quad (25)$$

taking the transpose of both sides of the first equation (B and W are symmetric)

$$v_1' B v_2 = \lambda_1 v_1' W v_2$$

thus

$$\lambda_1 v_1' W v_2 = \lambda_2 v_1' W v_2$$

$$(\lambda_1 - \lambda_2) v_1' W v_2 = 0$$

since $\lambda_1 \neq \lambda_2$, $v_1' W v_2 = 0$

therefore, $v_1' W v_2 = 0$ which means V_1 and V_2 are uncorrelated, and Y_2 has this property: its discriminant-criterion value, λ_2 , is the largest achievable by any linear combination of X 's that is uncorrelated (in the total sample) with Y_1 . Similarly

$$Y_3 = v_{31} X_1 + v_{32} X_2 + \dots + v_{3p} X_p \quad (36)$$

has the largest possible discriminant-criterion value (λ_3) among all linear combinations of the X 's that are uncorrelated with both Y_1 and Y_2 ; and so on until Y_r using the

elements of V_r as weights, has the largest possible discriminant-criterion value among linear combinations that are uncorrelated with all the preceding linear combinations $Y_1, Y_2, \dots, Y_{r-1}$. The linear combinations $Y_1, Y_2, \dots, Y_r$ are called the first, second, ..., r th (linear) discriminant functions for optimally differentiating among the g given groups.

The situation here is reminiscent of principal components analysis. There, the dimension corresponding to the first component had maximum variance; the second-component dimension had maximum variance among those uncorrelated with the first; and so on. In discriminant analysis, the ratio of between-to within-groups sums-of-squares merely takes the place of variance as the criterion in determining the successive dimensions. However, an important difference between the dimensions identified in discriminant analysis and those in component analysis is that the former are generally not mutually orthogonal in the test space, even though they are uncorrelated. That is, the axis representing the discriminant functions are not a subset of axes obtainable by rigid rotation of the original system of p axes; the discriminant rotation is an oblique rotation.

Just as in the principal components analysis, the dimensions represented by the discriminant functions may be interpreted meaningfully. Even if they are not, it may be possible to achieve parsimony by reducing the dimensionality of the space needed to describe group differences. In

seeking to interpret the discriminant functions, the goal is to determine which of the original p variables contribute most to each function. For this purpose, comparison of the relative magnitudes of the combining weights as given by the elements of each eigenvector of $W^{-1}B$ is inappropriate because these are weights to be applied to the variables in raw-score scales, and are hence affected by the particular unit used for each variable.

To eliminate the spurious effects of units of measurement on the magnitudes of combining weights, standardized variables should be used.

The relative magnitudes of these standardized weights may be assessed by multiplying each raw-score weight by the standard deviation of the corresponding variable as computed from the within-groups SSCP (Sum of Squares, Cross product) matrix. This amounts to multiplying each element of a given eigenvector V_m by the square root of the corresponding diagonal element of W . Thus, for each m , define

$$v_{mi}^* = w_{ii} v_{mi} \quad i = 1, 2, \dots, p \quad (27)$$

as the standardized discriminant weights. The relative contribution of the i th variable to the m th discriminant function may then be gauged by the magnitude of v_{mi}^* in comparison with the other weights v_{mj}^* .

Up to this point, it has been shown that the dimensionality of the discriminant space is equal to the number of

nonzero eigenvalues of $W^{-1}B$, which is the smaller of the two numbers, $G-1$ and p . It may often happen, that the number of significant discriminant dimensions may be even smaller. That is, not all of the discriminant function may represent dimensions along which statistically significant group differences occur.

C. SIGNIFICANCE TEST IN DISCRIMINANT ANALYSIS

A basic quantity in testing the significance of the overall difference among several group centroids (mean vectors) the ratio of the determinants of the within-groups and the total SSP matrices, known as Wilks' Λ criterion.

$$\Lambda = \frac{|W|}{|T|} \quad (28)$$

Motivation for use of this equation may be seen as follows:

$$\begin{aligned} \frac{1}{\Lambda} &= \frac{|T|}{|W|} = |W^{-1}T| \\ &= |W^{-1}(W + B)| \\ &= (1 + \lambda)(1 + \lambda_2); \dots, (1 + \lambda_r) \end{aligned} \quad (29)$$

where $\lambda_1, \lambda_2, \dots, \lambda_r$ are the nonzero eigenvalues of $W^{-1}B$. Consequently, Bartlett's V statistic for testing the significance of an observed value can be expressed as

$$\begin{aligned}
V &= - [N - 1 - (p + G)/2] \ln \Lambda \\
&= [N - 1 - (p + G)/2] \ln [(1 + \lambda_1)(1 + \lambda_r)] \quad (30) \\
&= [N - 1 - (p + G)/2] \sum_{m=1}^r \ln(1 + \lambda_m)
\end{aligned}$$

This statistic is distributed approximately chi-square with $p(G-1)$ degrees of freedom.

Because of the uncorrelatedness of the successive discriminant functions, the successive terms $\ln(1 + \lambda_m)$ in the last expression above are statistically independent (assuming multivariate normality of the original p variables). As a result, the additive components of V are each approximately distributed as a chi-square variate. More specifically, the m th component,

$$V_m = [N - 1 - (p + G)/2] \ln (1 + \lambda_m) \quad (31)$$

is approximately chi-square with $p + G - 2m$ degrees of freedom. It may be readily verified that the sum of the number of degree of freedom (n.d.f) of the r components, that is, $(p + G - 2) + (p + G - 4) + \dots + (p + G - 2r)$, is equal to $p(G - 1)$ regardless of whether $r = G - 1$ or p .

Consequently, when we cumulatively subtract V_1, V_2 , and so on from V , the remainder each time is also a chi-square variate; and these successive remainders become appropriate statistics for testing whether the residual discrimination after removing the first discriminant

function, the first and second discriminant function, and so forth, is statistically significant. The successive test statistics and their n.d.f.'s may be summarized as follows:

Residual After Removing	Approximate - Statistic	n.d.f.
First discriminant Function	$V - V_1$	$p(G-1) - (p+G-2)$ $= (p-1)(G-2)$
First 2 discriminant Function	$V - V_1 - V_2$	$(p-1)(G-2) - (p+G-4)$ $= (p-2)(G-3)$
First 3 discriminant Function	$V - V_1 - V_2 - V_3$	$(p-2)(G-3) - (p+G-6)$ $= (p-3)(G-4)$
⋮	⋮	⋮
First s discriminant Function	$V - V_1 - V_2 - V_3 \dots - V_s$	$(p-s)(G-(s+1))$

As soon as the residual, after removing the first s discriminant functions becomes smaller than the prescribed percentile point (that is, the $100(1 - \alpha)$ th percentile) of the appropriate chi-square distribution, we may conclude that only the first s discriminant functions are significant at that α level. If the number of significant discriminant functions thus found is smaller than r (as will often be the case), we will have effected a further reduction in the dimensionality of the space required to describe the differences among the G groups from which

our sample groups were drawn. The remaining $r-s$ dimensions may be regarded as immaterial for population differentiation, since our sample differences along these dimensions can be attributed to sampling error.

IV. CLUSTER ANALYSIS

A. ORIGIN AND THEORY

Clustering is the grouping of similar objects. The principal functions of clustering are to name, to display, to summarize, to predict, and to aid in interpretation of data with many dimensions. Clustering techniques were first developed in the field of biological taxonomy. It is one of several methodologies included in the broader category called classification.

The cluster analysis problem is the last step we consider in the progression of category sorting problems. While in discriminant analysis some part of the structure is known and missing information is estimated from labeled samples, the operational objectives of clustering is to classify new observations, that is, recognize them as members of one category or another. In cluster analysis little or nothing is known about the category structure. All that is available is a collection of observations whose category membership are known. We seek to discover a category structure which fits the observations. The problem may be stated as one of finding the "natural groups", which means to sort the observations into groups such that the degree of "natural association" is high among members of the same group and low between members of different groups.

Cluster analysis techniques have been applied in many fields of study. The literature is both voluminous and diverse, the terminology differing from one field to another. "Numerical taxonomy" is frequently substituted for cluster analysis among biologists, botanists, and ecologists, while some social scientists may refer "typology". Other frequently encountered terms are pattern recognition and partitioning. While discriminant analysis has been studied by statisticians for nearly 45 years, cluster analysis has only recently come to statistical notice. Any method which partition a set of objects into subsets on the basis of measurements taken on every object qualifies as a clustering method.

Most of the well known clustering techniques fall into one of two main categories: (1) hierachical and (2) non-hierachical (partitioning). The former is one in which every cluster obtained at any stage is a merger of clusters at previous stages. The nonhierachial procedures however form new clusters by lumping and splitting old ones. We consider both categories shortly.

In a geometric sense, every observation may be viewed as a point in p -dimensional Euclidean space. This swarm of data points may contain dense regions or "clouds" of data points which are separable from other regions containing a low density of points. These denser regions constitute what are known as clusters. In one and two dimensional cases, it is easy to visualize and to detect the clusters from scatter

plots, assuming that the clusters exist. In higher dimensions, clustering becomes extremely difficult without the aid of a computer.

Mathematical clustering techniques usually require a measure of similarity to be defined for every pairwise combination of the entities to be clustered. In order to solve the cluster problem, it is desirable to define the terms "similarity" and "difference" in a quantitative fashion. A researcher would assign two observations to the same group if the distance between them is sufficiently small, or to different clusters if this distance is sufficiently large.

At this point, two questions may be brought on. The first one is "how do we measure the distance between the observations?" and the second one is "how small is small enough?" and how large is large enough? These will be discussed in the following sections.

B. MEASURES OF DISTANCE

1. General

Let E_p be a symbolic representation for a measurement in p -dimensional space and let X, Y , and Z be any of these points in E_p . Then any nonnegative real-valued function $D(X, Y)$ satisfying the following conditions qualifies as a distance function (or metric).

1. $D(X, Y) = 0$ if and only if $X = Y$
2. $D(X, Y) \geq 0$ for all X and Y in E_p

$$3. \quad D(X,Y) = D(Y,X)$$

$$4. \quad D(X,Y) \leq D(X,Z) + D(Y,Z)$$

Many clustering algorithm assume such distances given and set about constructing clusters of objects within which the distances are small. The choice of distance function is no less important than the choice of variables to be used in the study. A serious difficulty in choosing a distance lies in the fact that a clustering structure is more primitive than a distance function and that knowledge of clusters changes the choice of distance function. Thus a variable that distinguishes well between two established clusters should be given more weight in computing distances than a "junk" variable that distinguishes badly.

2. Euclidean Distance

The Euclidean distance between the I-th and K-th observations of a data matrix X is defined as

$$D(I,K) = \left[\sum_{1 \leq J \leq p} \{X(I,J) - X(K,J)\}^2 \right]^{1/2} \quad (32)$$

where J is J-th variable. In one, two, or three dimensional space, this is just a "straight line" distance between the vectors corresponding to the I-th and K-th observations. When the variables are measured in different units, it is necessary to prescale the variabes to make their values comparable or, equivalently, to compute a weighted Euclidean distance.

$$D(I,K) = \left[\sum_{1 \leq J \leq p} W(J) (X(I,J) - (X(K,J)))^2 \right]^{1/2} \quad (33)$$

This form of distance is not necessary if all variables are measured on the same scale. However, even in this case, weights might be used to increase or decrease the importance of same variable. Various weighting schemes have been utilized in practice. One common weighting scheme lets $W(J)$ be the reciprocal of the variance of variable J .

A general class of squared distance functions is provided by utilizing positive definite quadratic forms. Specifically, if p represents a p -dimensional observation to be assigned to one of s groups, then to measure the squared distance between the observation β and the centroid (mean vector) of the i -th group one may consider the function

$$D_i = (\beta - \bar{x}_{i.})^T M (\beta - \bar{x}_{i.})$$

where M is a positive definite matrix to ensure that $D_i \geq 0$. Different distance functions are represented by different choices of the matrix M . When $M = I$ (the identity matrix) the resulting metric is the standard Euclidean distance. Distances with the Euclidean metric are shown in Figure 1a. The variance within the data may make the unweighted Euclidean metric inappropriate. As shown on the Figure 1b, where X has a larger variance than Y ,

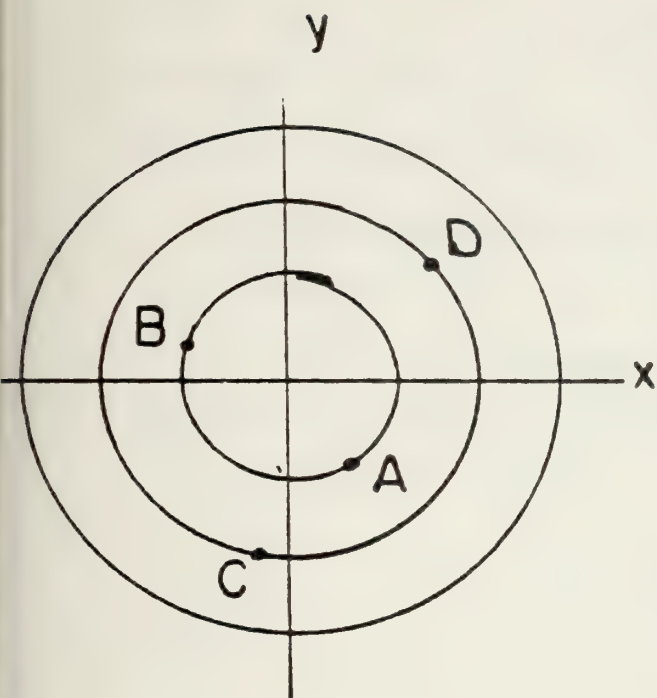
one may wish to weight a deviation in the X direction less than an equal deviation in the Y direction. This is a weighted Euclidean distance function which makes point A and B equidistance from the origin. In this case, the matrix M is diagonal elements which are the reciprocals of the variances of the different variables.

Extending this idea further, it may be possible to consider the covariance among variables as well. Figure 1c shows how the axis may be rotated so that the major axis is oriented in a direction of reflecting the positive correlation between X and Y. Again, points on the same ellipse are considered equidistance from the origin. The matrix M in this case is the inverse of the covariance matrix.

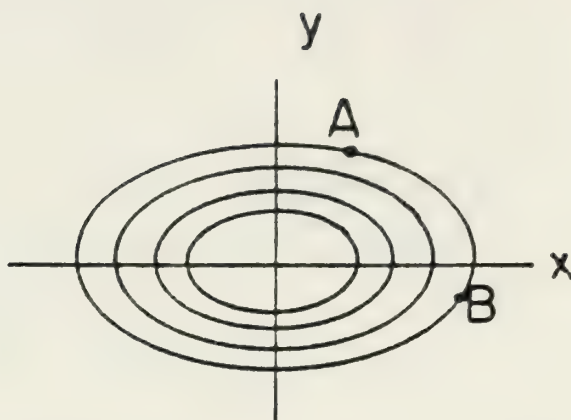
Further extension of this concept will explain some sort of generalized distance function. If C_i represents the covariance matrix of the i th cluster then the distance function

$$D_i = (\beta - \bar{x}_{i.})^T C_i^{-1} (\beta - \bar{x}_{i.})$$

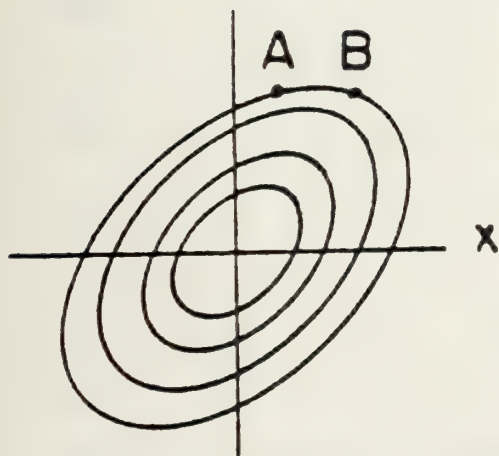
uses the appropriate covariance structure when determining the distance to a particular cluster centroid. Since C_i changes to reflect the dispersion internal to each particular cluster, the use of this metric exploits differences in the dispersion characteristics of the different groups. As shown on Figure 1d, not how a new observation (denoted by u) is



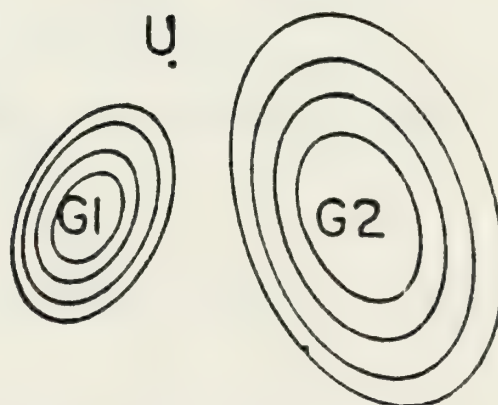
1a. Euclidean measure of squared distance.



1b. Measure of squared distance with different weights for variables.



1c. Generalized squared distance measure.



1d. Classification when within-group dispersions are different.

Figure 1. Euclidean Distance

closer to the centroid of group one (G1) in terms of Euclidean distance but is more likely to be assigned to group two (G2) when using the C_i matrix.

3. Mahalanobis Distance

Another choice for the M matrix in equation (1) is p^{-1} where P represents the pooled within groups covariance matrix of all the clusters.

$$P = \frac{1}{\sum_{i=1}^G (n_i - 1)} W \quad (34)$$

where

$$W = \sum_{k=1}^G W_k$$

This distance is the well known Mahalanobis distance. Note that P does not change from group to group. To ensure the non-singularity of P it must be true that $p \leq (N - G)$, where N represents the total number of observations over all groups. Rewriting the distance,

$$D_i = (\beta - \bar{x}_{i.})^T W^{-1} (\beta - \bar{x}_{i.}) \quad (35)$$

defines a distance between mean vectors β and $\bar{x}_{i.}$ and common covariance matrix W . The Mahalanobis distance function adjusts for both scale of measurement of the variables and covariation among the variables. Use of this

metric is equivalent to computing distances on variables transformed to their principal components. This metric is invariant under any nonsingular transformation of original variables. For consider the transformation

$$\dot{Y} = BX \quad (36)$$

and let $D(Y_i, Y_j)$ represent Mahalanobis distance between Y_i and Y_j .

$$\begin{aligned} D(Y_i, Y_j) &= (Y_i - Y_j)^T P_Y^{-1} (Y_i - Y_j) \\ &= (BX_i - BX_j)^T P_{\dot{Y}}^{-1} (BX_i - BX_j) \\ &= (X_i - X_j)^T B^T P_Y^{-1} B (X_i - X_j) \\ &= (X_i - X_j)^T B^T (BP_X B^T)^{-1} B (X_i - X_j) \\ &= (X_i - X_j)^T P_X^{-1} (X_i - X_j) \\ &= D(X_i, X_j) \end{aligned}$$

Some other common metrics are listed below:

1. L_1 norm (City Block)

$$D(X_i, X_j) = \sum_{k=1}^P |X_{ki} - X_{kj}|$$

2. L_p norm (Minkowsky Metrics)

$$D(X_i, X_j) = \left(\sum_{k=1}^P |X_{ki} - X_{kj}|^p \right)^{1/p}$$

3. Uniform norm

$$D(X_i, X_j) = \text{Superemum}_{k=1, 2, \dots, p} \{ |x_{ki} - x_{kj}| \}$$

C. HIERARCHICAL CLUSTERING

1. General

The previously discussed distance measures may be used to construct a similarity matrix describing the length of all pairwise relationships among the entities (variables or data units) in the data set. The methods of hierachical cluster analysis operate on this similarity matrix to construct a tree depicting specified relationships among the entities. As shown on Figure 2, the branches on the left each represent one entity while the root represents the entire collection of entities. Moving down the tree from the branches toward the root depicts increasing aggregation of the entities into clusters. Hierarchical clustering methods which build a tree from branches to root often are called agglomerative methods.

Once a tree is constructed for N entities, the analyst may choose from as many as N sets of clusters. These clusters are nested. From the agglomerative view, when two entities are merged they are joined together permanently and considered as one entity for later merges; from the divisive view, when a group of entities is split into two parts, the parts are separated permenently and may be treated independently for the remainder of the analysis.

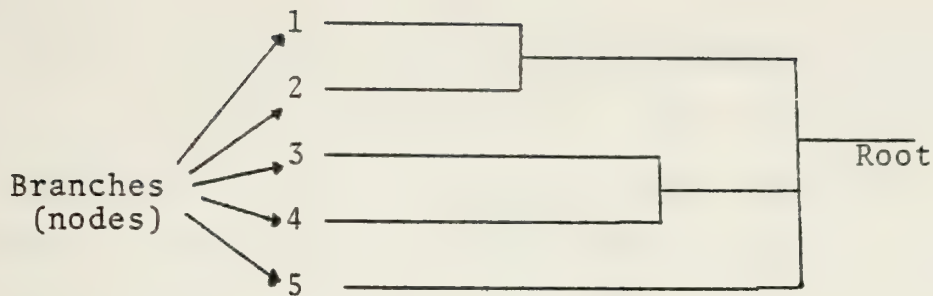


Figure 2. Tree for Hierarchical Clustering

Herein lie both the strength and weakness of hierarchical methods: by taking early decisions as permanent, the number of possibilities that need be examined is reduced greatly as compared with complete enumeration; but this same convention precludes discovering early mistakes or capitalizing on later opportunities.

There are three major hierarchical clustering concepts:

1. Linkage Methods
2. Centroid Methods
3. Error sum of squares or variance methods.

All of these methods are suitable for clustering data units. However, only the linkage methods are considered in this research.

2. The General Agglomerative Procedure

Let s_{ij} be the similarity between entities i and j as defined by one of the distance measures previously discussed. Assuming that the similarity is symmetric, the complete schedule of similarities for all $\binom{N}{2} = \frac{1}{2}N(N - 1)$

possible pairwise combinations of entities may be arrayed in a lower triangular similarity matrix as in Figure 3. The s_{ij} entries are nonnegative. This limitation is of consequence only for correlation and the cosine of the angle between vectors; the distinction between positive and negative association cannot be utilized in these clustering methods.

$$S = \begin{array}{c|cccc} & s_{21} & & & \\ & s_{31} & s_{32} & & \\ & s_{41} & s_{42} & s_{43} & \\ & . & . & . & \\ & . & . & . & . \\ & . & . & . & . \\ s_{n1} & s_{n2} & s_{n3} & \cdots & s_{n(n-1)} \end{array}$$

Figure 3. Lower Triangular Similarity Matrix

A simple remedy is to use the absolute value or the square of the measure if it can assume negative values. Once the matrix is defined, the process of clustering entities is almost trivially simple. The general procedure for agglomerative clustering on a data matrix is as follows:

- (1) Begin with n clusters each consisting of exactly one entity. Let the clusters be labeled with the numbers 1 through N .

- (2) Search the similarity matrix for the most similar pair of clusters. Let the chosen clusters be labeled p and q and let their associated similarity be s_{pq} , $p > q$.
- (3) Reduce the number of clusters by 1 thorough merger of clusters p and q . Label the product of the merger q and update the similarity matrix entities in order to reflect the revised similarities between cluster q and all other existing clusters. Delete the row and column of S pertaining to cluster p .
- (4) Perform steps 2 and 3 a total of $N-1$ times (at which point all entities will be one cluster). At each step record the identity of the clusters which are merged and the value of similarity between them in order to have a complete record of the results.

Different agglomerative methods are implemented by varying the procedures used for defining the most similar pair at step 2 and for updating the revised similarity matrix at step 3. The similarity matrix is a given array of numbers. The numerical execution of the clustering procedures is completely independent of how the similarity values were generated or whether the entities to be clustered are variables or data units. However, it is necessary to make a direct distinction between distance-like

measures (the smallest values correspond to the most similar pairs) and correlation-like measures (the largest values correspond to the most similar pairs); the essential difference is whether the search for the most similar pair involves seeking the minimum or maximum entry in the similarity matrix.

3. Single Linkage

The method of single-linkage cluster analysis is the simplest of all hierarchical techniques. At each stage, after clusters p and q have been merged, the similarity between the cluster (labeled t) and some other r is determined as follows:

1. If s_{ij} is the distance-line measure

$$s_{tr} = \min (s_{pr}, s_{qr}) \quad (37)$$

The quantity s_{tr} is the distance between the two closest members of clusters t and r . If clusters t and r were to be merged, then for any entity in the resulting cluster the distance to its nearest neighbor would be at most s_{tr} .

2. If s_{ij} is a correlation-like measure

$$s_{tr} = \max (s_{pr}, s_{qr}) \quad (38)$$

The quantity s_{tr} is the similarity between the two most similar entities in clusters t and r . If clusters t

and r were to be merged, then for any entity in the resulting cluster there would be at least one other entity in the same cluster such that the pair would have a similarity at least as large as s_{tr} .

The method is known as single linkage because clusters are joined at each stage by the single shortest or strongest link between them. Since the updating process involves choosing only the minimum or maximum single-linkage clustering is invariant to any transformation which leaves the ordering of the similarities unchanged; that is, any monotonic transformation.

4. Complete Linkage

The complete-linkage method is related to the single-linkage method and is no more difficult to execute. At each stage, after clusters p and q have been merged, the similarity between the new cluster (labeled t) and some other cluster r is determined as follows:

1. If s_{ij} is distance-like measure

$$s_{tr} = \max (s_{pr}, s_{qr}) \quad (39)$$

The quantity s_{tr} is the distance between the most distant members of clusters t and r . If clusters t and r were merged, then every entity in the resulting cluster would be no farther than s_{tr} from every other entity in the cluster. The value of s_{tr} is the diameter of the smallest sphere which can enclose the cluster resulting from the merger of clusters t and r .

2. If s_{ij} is a correlation-like measure

$$s_{tr} = \min (s_{pr}, s_{qr}) \quad (40)$$

The quantify s_{tr} is the similarity between the two most dissimilar entities in clusters t and r . If clusters t and r were to be merged, then every entity in the resulting cluster would have a similarity of at least s_{tr} with every other entity in the cluster.

The method is called complete linkage because all entities in a cluster are linked to each other at some maximum distance or minimum similarity. Such a cluster is called a "maximally connected subgraph" in graph theory. In contrast to the single-linkage method, interpretation of the clusters can be made only in terms of the relationships within individual clusters; there is no particularly useful interpretation involving the differences between clusters. Like the single-linkage method, complete-linkage cluster analysis is invariant to monotonic transformations of the similarity measure. Johnson (1967) discusses this property in both single and complete linkage methods.

D. NONHIERARCHICAL CLUSTERING

Nonhierarchical clustering methods are designed to cluster data units into a single classification of g clusters, where g either is specified a priori or is determined as a part of the clustering method. The central idea in most of these methods is to choose some initial

partition of the data units and then alter cluster memberships so as to obtain a better partition. The various algorithms which have been proposed differ as to what constitutes a "better partition" and what methods may be used for achieving improvements.

The broad concept for these methods is very similar to that underlying the steepest descent algorithms used for unconstrained optimization in nonlinear programming. Such algorithms begin with an initial point and then converge to a local optimum, moving one step at a time, the value of the objective function improving at each step.

The methods of nonhierarchical clustering typically may be used with much larger problems than the hierarchical methods because it is not necessary to calculate and store the similarity matrix; it is not even necessary to store the data set. In general, the data units are processed serially and can be read from tape or disk as needed. This characteristic makes it possible, at least in principle, to cluster arbitrary large collections of data units.

In this research, we consider only the partitioning method known as "K-MEANS" which was developed by MacQueen (15). He used the term "K-MEANS" to denote the process of assigning each data unit to that cluster (of k clusters) with the nearest centroid (mean vector). The cluster centroid changes with each transfer of an observation.

The decomposition of the total scatter matrix into within and between groups matrices suggests possible

optimality criteria to be used in a clustering algorithm. One would like the within-groups scatter to be small relative to the between-groups scatter. Various trial clusterings could be formed using the W and B matrices as a basis for the optimality criteria which determine the best clustering. A possible choice for a criterion is to minimize trace W over all partitions into g groups. Since T is constant over all partitions, minimizing trace W is equivalent to maximizing traces B since

$$\text{trace } T = \text{trace } W + \text{trace } B \quad (41)$$

Although trace W is invariant under an orthogonal transformation, it is not invariant under other non-singular linear transformations.

McRae (16) points out that trace W equals the total within group sum of squares, hence the "minimum variance partition" cluster solution is found by minimizing trace W .

Considerable study has been developed to alternative criteria such as those based on multivariate statistical analysis techniques, especially the methods of linear discriminant analysis and multivariate analysis of variance. Assuming the p variables are not linearly dependent, then as long as $p = N - g$, W is positive definite symmetric and so is W^{-1} . Attempts to make B and W as different as possible lead one to solving the determinantal equation:

$$|B - \lambda W| = 0 \quad (42)$$

The solutions λ_i are the eigenvalues of the matrix $W^{-1}B$ as in discriminant analysis. There are t non-zero eigenvalues, where t is the minimum of p and $g-1$. This is a consequence of the fact that, if g is less than p , the g group means are considered in a $(g-1)$ -dimensional hyperplane. When $g = 2$ the analysis is equivalent to two-group discriminant analysis. Linear discriminant analysis would take the vectors originally described in p -dimensional coordinate system and transform the basis to a t -dimensional system. Maximizing the largest of these eigenvalues is a criterion suggested by S.N. Roy and maximizing the trace of $W^{-1}B$, however is a criterion suggested by Hotelling. In both cases, large values for these statistics are sought in clustering algorithms since large values indicate large differences among (between) groups. Minimizing the ratio of determinants $|W| \div |T|$ is a criterion widely known as Wilks' lambda discussed in the discriminant analysis. Since T is the same for all partitions, this criterion is equivalent to minimizing determinant W . Both trace $W^{-1}B$ and $|T| \div |W|$ may be expressed in terms of the eigenvalues of $W^{-1}B$.

$$\left| \frac{T}{W} \right| = \prod_{i=1}^t (1 + \lambda_i) \quad (43)$$

$$\text{trace } W^{-1}B = \sum_{i=1}^t \lambda_i \quad (44)$$

where $t = \min(p, g-1)$. Therefore minimizing $\det W$ is equivalent to maximizing $\pi(1 + \lambda_i)$.

Friedman and Rubin (6) describe the advantages of the various criteria. Those based on multivariate statistical considerations (all but trace W) are invariant under changes in scale for variables (non-singular linear transformation). In fact, they are the only invariants for W and B under such transformations. In addition, the multivariate criteria may take into account covariation among the variables.

V. ANALYSIS OF MULTIVARIATE UTILITY DATA

To illustrate hierarchical clustering we applied the technique described in the previous chapter to partition a set of twenty six attributes of a close-air support weapon system into a smaller collection of "superattributes". As part of an effort to evaluate the military utility of a proposed alternative U.S. Marine Corps air support rada system, AN-TPQ/27. Barr and Richards (4) extracted 26 attributes of the TPQ-27 and a baseline system, the AN-TPQ/10, and then had members of the Operational Test and Evaluation Team assess the utility of the TPQ/27 relative to that of the TPQ/10. In order that the additive model used to combine unidimensional relative utilities into a system relative utility be justifiable, it is necessary that the utilities satisfy certain independence properties described in Keeney and Raiffa (12).

Because those independence properties are very difficult for decision makers to verify for complex alternatives like the weapon systems under study, Professors Barr and Richards attempted instead to work with the attributes to try to generate a new collection which would likely satisfy, at least approximately, the conditions required to justify the additive model.

The original collection of 26 attributes is as follows:

1. Portability
2. Durability
3. Time to Set Up
4. Time to Take Down
5. Ease of Assigning Aircraft to Targets
6. Number of Aircraft Controlled
7. Number of Targets
8. Communications
9. Mission Flexibility
10. ASRT Survivability
11. Time to Locate and Acquire Aircraft
12. Accuracy of Tracking
13. Accuracy of Delivery
14. Range
15. Aircraft Vulnerability
16. Aircraft Attack Throughout
17. Base of Adjustment and Evaluation of Results
18. Accuracy of Feedback
19. Ease of Operation
20. Man-Machine Compatibility
21. Training Requirements
22. Reliability
23. Maintainability
24. Supportability
25. Availability
26. Documentation

where a_i represents an attribute i and

$$I(x_{ij}, x_{kj}) = \begin{cases} 1 & \text{if } x_{ij} = x_{kj} \\ 0 & \text{if } x_{ij} \neq x_{kj} \end{cases}$$

It is easy to verify that D is a metric as defined in Chapter IV. Since we will actually work with a similarity measure in the hierarchical cluster procedure, we define the similarity between two attributes a_i and a_k as

$$S(a_i, a_k) = \sum_{j=1}^{12} I(x_{ij}, x_{kj}) \quad (46)$$

One can see from this definition that the similarity between two attributes a_i and a_k is simply the number of team members who placed attributes a_i and a_k in the same partition. For example,

$$\begin{aligned} S(a_1, a_2) &= 0 + 1 + 0 + 1 + 0 + 0 + 0 + 1 + 0 + 0 + 0 \\ &\quad + 1 + 0 = 4 \end{aligned}$$

Either S or D can be used in the computer program shown in Appendix A for hierarchical clustering. One need only indicate whether he wants a correlation-like (larger values imply more similar) measure or a distance-like measure (smaller values imply more similar). We selected to use the former method. The similarity matrix extracted from

Table II. Data Matrix

	1	2	3	4	5	6	7	8	9	10	11	12
1	6	2	1	3	2	4	3	1	3	1	1	1
2	1	2	2	3	1	3	2	1	1	5	1	2
3	6	1	1	5	2	4	3	1	6	1	2	1
4	6	1	1	5	2	5	3	1	6	1	2	1
5	2	5	3	1	3	1	1	2	2	2	3	3
6	2	7	4	1	4	1	1	2	4	2	3	3
7	2	7	4	1	4	1	1	2	4	2	3	3
8	3	5	4	7	5	6	1	3	1	3	3	6
9	2	5	4	1	3	1	1	2	4	2	3	3
10	9	6	5	6	5	7	8	3	7	3	2	4
11	2	8	6	4	3	2	1	4	4	2	3	3
12	3	8	7	4	4	2	6	5	4	6	4	3
13	3	8	7	4	4	2	6	5	4	6	4	3
14	3	8	4	4	4	2	6	5	4	2	4	3
15	7	6	5	6	5	7	7	3	7	3	7	4
16	8	8	6	1	4	1	7	4	4	2	7	3
17	8	8	8	4	3	2	5	6	5	4	5	3
18	4	8	8	4	6	2	5	6	5	6	5	3
19	4	5	3	1	3	1	1	2	2	4	3	3
20	4	5	3	3	3	3	3	1	2	4	8	3
21	5	4	9	2	3	8	4	7	2	4	6	5
22	1	3	2	3	1	3	2	7	1	5	6	2
23	1	3	9	3	1	3	2	7	1	5	6	2
24	1	3	4	3	1	3	2	7	3	5	6	2
25	1	3	2	3	1	3	2	7	1	5	6	2
26	5	4	9	2	6	8	4	7	2	4	6	5

from the data is shown in Table V-3. We present only lower triangular elements since $S(a_i, a_i) = 12$ for all i and the matrix is symmetric; i.e., $S(a_i, a_k) = S(a_k, a_i)$. Zero values are not written.

The results from the hierarchical clustering are shown in Figure 4. The numbers printed along the left hand margin refer to the attribute numbers. As you proceed to the right through the tree you will observe numbers greater than 26. These correspond to the clusterings that takes place from one step to the next. For example, the number 27 shown at the juncture of 25 and 22 means that the first attribute clustered together should be 25 and 22 (this is the most similar pair). This combination is then considered as a new attribute which is later combined with the attribute 30 (itself a combination of 23 and 24) to form the attribute 31. This is later combined with attribute 2 to form attribute 40, etc.

As discussed in Chapter IV a decision has to be made as to how many clusters (superattributes) are desired. All hierarchical methods will continue clustering until there is a single cluster. In order to decide on the number of clusters (and their composition) one need only image drawing a vertical line through the tree at various places. Each intersection of the tree with the vertical line results in a cluster. For example, the vertical line at the point A results in the 6 clusters shown in Table V-4.

It is clear from observing the above collection that some of the attributes are highly correlated and nonredundant. If one tries to assign an importance weights to each attributes separately, there is a distinct likelihood that some of the overlapping strongly into related attributes might effectively be double or triple weighted or more producing biased result. It is an effort to prevent this from happening, Barr and Richards aksed the utility assessment team to partition the 26 attributes into a smaller collection in such a way that attributes within a group are similiar and attributes in different groups are unrelated the sense that utility assessments for attributes in one group do not depend on the amounts of attributes in any other group.

The total number of groups was not prespecified. Instead, each team member was allowed to partition the 26 attributes into any number of groups. The resulting multivariate data array is shown in Table V-2. An element x_{ij} is the number of the group into each team member j put attribute i .

Let us define a distance measure for this data array as follows:

$$D(a_i, a_k) = \sum_{j=1}^{12} (1 - I(x_{ij}, x_{kj})) \quad (45)$$

Table III. Similarity Matrix for Superattribute Determination

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
1																										
2	4																									
3	8	1																								
4	7	1	11																							
5																										
6					8																					
7					8	12																				
8		1			3	3	3																			
9					10	10	10	3																		
10			1	1				3																		
11					6	6	6	2	7																	
12					1	3	3	3	1	2		5														
13					1	3	3	3	1	2		5	12													
14					2	5	5	2	4			5	9	9												
15								3		9																
16					3	5	5		4	1	6	3	3	4	3											
17					2	1	1	1	2		5	4	4	3		2										
18					1	1	1	1	1		5	4	4	3		1	9									
19					9	5	5	2	7		3					2	2	1								
20	2	3	1	1	5	1	1	1	3	2						2	1	8								
21					1				1	1						2		3	3							
22	2	8						1												2	2					
23	2	7						1												2	3	11				
24	3	6						1												2	3	10	11			
25	2	1																		2	2	12	11	10		
26					1												1	1	2	2	9	3	4	4	3	

The superattributes used in the utility study are those shown in Table IV. A careful examination of the attributes which compare the clusters shows that the results so obtained are intuitively agreeable. The names supplied to the superattribures are somewhat natural descriptions of the clusters obtained.

The listing of the computer program and sample output are given in Appendix A.

Table IV. Superattributes

<u>Superattributes</u>	<u>Component Attributes</u>
Facility of movement	1. Portability 3. Time to set up 4. Time to take down
-----	-----
Facility of Use (precision)	5. Ease of assigning aircraft to targets 6. Number of aircraft controlled 7. Number of targets 9. Mission flexibility 11. Time to locate and acquire aircraft 16. Aircraft attack throughput 17. Ease of adjustment 18. Accuracy of feedback 19. Ease of operation 20. Man-machine compatibility 12. Accuracy of tracking 13. Accuracy of delivery 14. Range
-----	-----
Survivability	10. ASRT Survivability 15. Aircraft vulnerability
-----	-----
Learning	21. Training requirements 26. Documentation
-----	-----
Readiness	2. Durability 22. Reliability 23. Maintainability 24. Supportability 25. Availability
-----	-----
Communications	8. Communications
-----	-----

VI. ANALYSIS OF ARMY TANK DATA

A. DATA STRUCTURE

In order to illustrate the nonhierarchical clustering methodology, principal components analysis, and discriminant analysis data on Army tanks from eight different countries were taken from Jane's Book of Weapon Systems (1979-80). A total of twenty-four tanks were included in the data array with observation on each of 10 variables. The 10 variables are listed below:

1. Weight (ton)
2. Length (meter)
3. Width (meter)
4. Height (meter)
5. Road Speed (kilometer per hour)
6. Trench Crossing (meter)
7. Ground Pressure (Kg/cm^2)
8. Maximum Armament (rounds)
9. Ground Clearance (meter)
10. Power to Engine Ratio (BHP/ton)

The twenty-four tanks and the associated countries are shown below:

Identification Number	Type/Name	Country
11	T-62	
12	T-54	U.S.S.R.

Identification Number	Type/Name	Country
13	T-10	
14	ASU-85	
15	MK-5/Chieftain	
16	MK-3/Vickers	
17	MK-13/Centurion	U.K.
18	CVR(T)/Scorpion	
19	XM-1	
20	M60A2	
21	M60	U.S.A.
22	M48	
23	M47	
24	PZ61	
25	PZ68	SWITZER- LAND
26	STRV-103	
27	Ikv-91	SWEDEN
28	TYPE61	
29	TYPE74	JAPAN
30	Leopard 2	
31	Leopard	W. GER- MANY
32	TAM	
33	AMX 30	
34	AMX 13	FRENCH

We conjecture that a cluster analysis of the tank data will result in clusters corresponding to nationality since the nations may have different emphasis on the variables in the design of their tanks.

B. NONHIERARCHICAL CLUSTER ANALYSIS OF TANK DATA

1. The MIKCA Algorithm

The specific algorithm chosen for the nonhierarchival cluster analysis for the tank data is the MIKCA (Multivariate Iterative K-MEANS Clustering Algorithm) program written by Douglas J. McRae as a part of his doctoral dissertation at the University of North Carolina, Chapel Hill.

Reference to the flow chart in Figure 5 will aid the reader in following discussion of the algorithm. Inputs to program are the data matrix, an estimate for g (the number of clusters), and choice of criterion and distance functions.

In the first step, preliminary calculations are made, such as the variable means and standard deviations, as well as the cross product matrix T . The next step forms the initial cluster centers. Then each of the other observations is assigned to the nearest cluster. Euclidean distance is used for this initial phase, and the cluster centroids are recomputed after each observation is assigned to a group. The observations are considered in the same order as they were input. After all of them have been assigned to clusters, the criterion value is computed. This initial cluster-finding technique is referred to as a

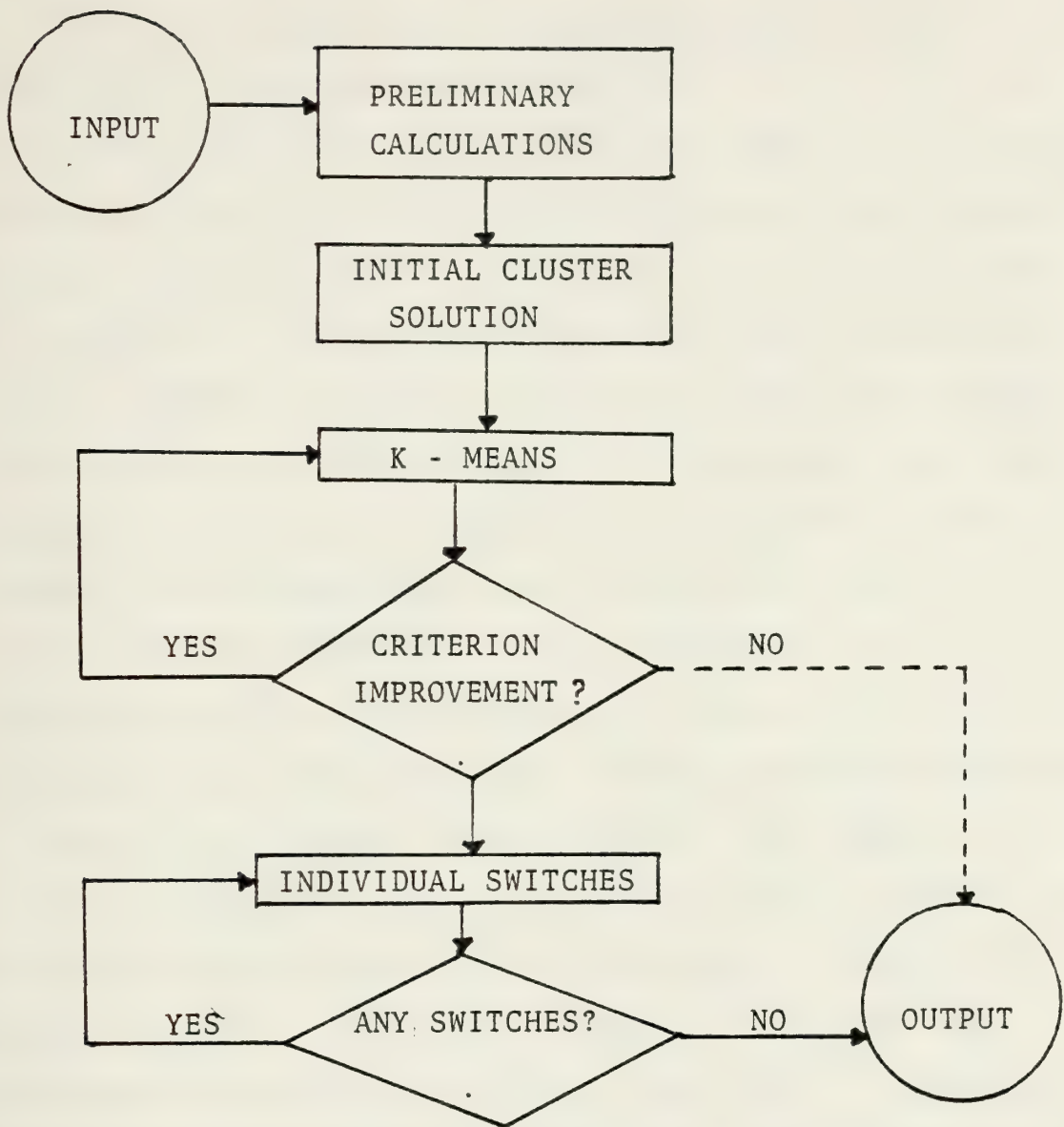


Figure 5. MIKCA Flow Chart

one-pass K-MEANS procedure. It is performed three times, and the solution which yields the best criterion value is chosen as the initial cluster solution.

After the initial solution has been found, the program advances to the iterative K-MEANS phase where the observations are again considered in the order in which they were input to the program. It is this phase where the user's choice of distance function is used. The distance from each observation to each cluster centroid is again computed, this time with the user's distance function, the assignment to the closest centroid being made and the centroid updated to reflect its new membership. After considering all n observations in this manner, the new criterion value is checked for possible improvement during the K-MEANS iteration. As long as the criterion value improves, the K-MEANS procedure is repeated; if the criterion fails to improve then the MIKCA algorithm goes to the next step, the individual switches section. Note the importance of the order of consideration of the observations. The order is important because the cluster means are recomputed after each observation is reassigned.

In the individual switches phase, consideration is given to moving each observation to every other cluster, the move being made if and only if an improvement in the value of the criterion results. An elaborate labelling procedure provides a unique order in which to consider each observation.

This procedure continues until a complete pass through the data is made with no changes in cluster membership.

The MIKCA algorithm provides the following options for distance and criterion functions.

Criterion

1. Minimum trace W
2. Minimum determinant W
3. Maximum largest order of $|B - \lambda W| = 0$
4. Maximum sum of roots of $|B - \lambda W| = 0$

Distance

1. Euclidean
2. Weighted Euclidean
3. Mahalanobis

A complete computer program is listed in Appendix B.

2. Cluster Results for Tank Data

For clustering of the tank data we selected the minimum trace W criterion and the weighted Euclidean distance function. The algorithm automatically provides weights for the weighted Euclidean distance function. The results of the clustering with four clusters are shown in Table V.

The conjecture of clustering by nationalities is supported by the results. The three Soviet tanks make up one cluster and the two British and four of the United States tanks were found to be similar. A third cluster consists of four tanks which are very lightweight. The

Table V. THE FINAL CLUSTER SOLUTION IS

CLUSTER	SIZE	CENTER	THE OBSERVATIONS ARE	7.00	3.26	2.55	59.18
1	11						
		40.60	2.64	0.86	53.64	0.46	20.70
		19	24	25	26		
		28	30	31	32		
		33					

CLUSTER	SIZE	CENTER	THE OBSERVATIONS ARE	7.15	3.55	3.06	43.05
2	6						
		51.87	2.80	0.86	59.50	0.43	15.95
		17	20	21	22		
		15					
		23					

CLUSTER	SIZE	CENTER	THE OBSERVATIONS ARE	5.40	2.77	2.21	64.37
3	4						
		13.08	2.14	0.91	45.75	0.38	20.07
		18	27	34			
		14					

CLUSTER	SIZE	CENTER	THE OBSERVATIONS ARE	6.85	3.50	2.25	44.33
4	3						
		41.00	2.85	0.77	37.67	0.44	15.07
		12	13				
		11					

final cluster consists of the rest of the tanks, including tanks of United States allies from West Germany, France, Sweden, Switzerland and Japan.

A natural question to ask after observing the results of a cluster analysis is what variables most strongly influence the clustering that was observed. A clue is provided by the composition of the cluster containing all of the lightweight tanks. This suggests that weight is an important distinguishing feature. This is examined in the principal components analysis and the discriminant analysis in the next two section.

C. PRINCIPAL COMPONENTS ANALYSIS

The Statistical Package for Social Sciences (SPSS) (14) subprogram FACTOR was used for the principal components analysis. It is designed both for the factor analysis and the principal components analysis. The outputs are designed to be self-explanatory. In this example, the first 5 components accoung for 90% of the variance and the remaining components account for only 10% of the variance (Figure 6).

The subprogram FACTOR provides a graphical presentation (Figure 7) for the factors that have been determined by the orthogonal rotations (in this example, variance maximization rotation). In reading the graphs, one should be attentive to following three features: (1) the relative distance of a variable from the axis, (2) the direction of a variable in relation to the axis, and finally (3) clustering of variables and their relative position to each other.

PRINCIPAL COMPONENT ANALYSIS OF TANKS
 FILE NCN/PI (CREATION DATE = 03/19/80)

03/19/80

VARIABLE	EST COMMUNALITY	FACTOR	EIGENVALUE	PCT OF VAR	CLM PCT
X1	1.00000	1	4.78856	47.5	47.5
X2	1.00000	2	1.71316	17.1	65.0
X3	1.00000	3	1.26768	12.7	77.7
X4	1.00000	4	0.74450	7.4	85.1
X5	1.00000	5	0.55505	5.6	90.7
X6	1.00000	6	0.34472	3.4	94.1
X7	1.00000	7	0.26042	2.6	96.7
X8	1.00000	8	0.19890	2.0	98.7
X9	1.00000	9	0.08262	0.8	99.6
X10	1.00000	10	0.04438	0.4	100.0

Figure 6. Summary Table of Principal Components Analysis on Tank

PRINCIPAL COMPONENT ANALYSIS OF TANKS
 FILE ACAPPE (CREATION DATE = 03/15/80)

03/15/80

HORIZONTAL FACTOR 1 VERTICAL FACTOR 2

1 = XX
 2 = XX
 3 = XX
 4 = XX
 5 = XX
 6 = XX
 7 = XX
 8 = XX
 9 = XX
 10 = XX

1 = XX
 2 = XX
 3 = XX
 4 = XX
 5 = XX
 6 = XX
 7 = XX
 8 = XX
 9 = XX
 10 = XX

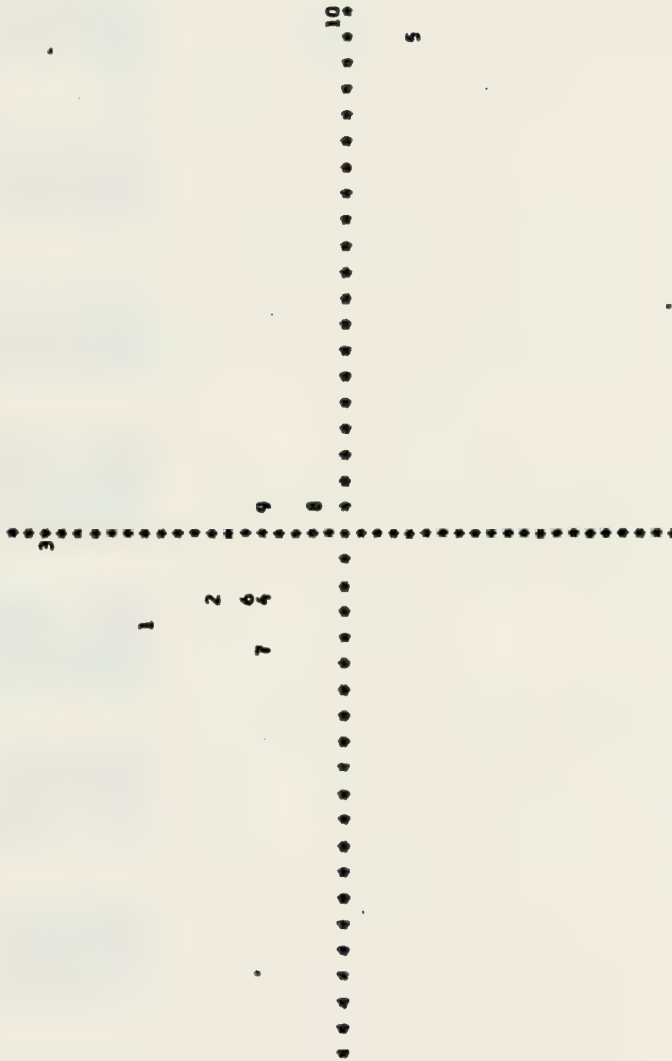


Figure 7. Graphical Presentation

For example, variables 5 (road speed) and 10 (power to engine ratio) contribute heavily to the first principal component while variables 1 (weight) and 3 (width) contribute most strongly to the second principal component. Variables 2, 4, 6, 7, 8, 9 are not as important. The weights accorded each variable in the 10 factors (principal components) are shown in Figure 8. The complete SPSS program is listed in Appendix C.

D. DISCRIMINANT ANALYSIS

The SPSS subprogram DISCRIMINANT was used to determine that function or those functions of the 10 variables that best discriminant among the four clusters determined in previous section.

The maximum number of discriminant functions to be derived is either one less than the number of groups or equal to the number of discriminating variables. This subprogram provides two measures for judging the importance of discriminant functions. One of these is the relative percentage of the eigenvalue associated with the function. It is a measure of the relative importance of the function. The sum of the eigenvalues is a measure of total variance existing in the discriminating variables. Since discriminant functions are derived in order of their importance, this process can be stopped whenever the relative percentage is judged to be too small. Of course, there is no fixed rule for deciding what is too small. In this research, we selected arbitrary, a significance level of 0.10. The output shown

in Figure 9 suggests that we therefore consider only the first two discriminant functions.

The second measure judging the importance of a discriminant function is its associated canonical correlation. The canonical correlation is a measure of association between the single discriminant function and the set of $(g-1)$ dummy variables which define the g group memberships. It tells us how closely the function and the group variable are related, which is just another measure of the function's ability to discriminate among the groups. From Figure 10, the first two discriminant functions are each highly correlated with the groups but the third has only a moderate correlation.

The next criterion for eliminating discriminant functions is to test for the statistical significance of discriminating information not already accounted for by the earlier functions. As each function is derived, starting with no (zero) functions, Wilks' lambda is computed. Lambda is an inverse measure of the discriminating power in original variables which has not yet been removed by the discriminant functions - the larger lambda is, the less is the information remaining. Lambda can be transformed into a chi-square statistic for an easy test of statistical significance. In Figure 9, Wilks' lambda was .594 after the first two functions had been derived. This corresponds to a chi-square of 8.8476 with a probability level of .1823.

TANK DISCRIMINANT

03/15/60

SUMMARY TABLE

STEP	ACTION ENTERED	VAR IN	WILKS' LAMBDA	SIG. D SQUARE	SIG.	BETWEEN GROUPS	LABEL
1	1	1	0.54693	0.0011	0.73235	2	MAX ARMAMENT
2	2	2	0.00000	0.00000	2.27114	2	TRENCH CROSSING
3	3	3	0.00000	0.00000	4.73511	1	WEIGHT
4	4	4	0.00000	0.00000	5.67933	1	WEIGHT CLEARANCE
5	5	5	0.00000	0.00000	6.8135	2	WIDTH
6	6	6	0.00000	0.00000	7.4106	2	WEIGHT PRESSURE
7	7	7	0.00000	0.00000	7.6838	2	ROAD SPEED
8	8	8	0.00000	0.00000	10.409	2	POWER TO ENGINE
9	9	9	0.00000	0.00000	13.219	2	RATIO
10	10	10	0.00000	0.00000	11.032	2	WEIGHT CLEARANCE
11	11	11	0.00000	0.00000	11.032	2	WEIGHT CLEARANCE
12	12	12	0.00000	0.00000	11.032	2	WEIGHT CLEARANCE

CLASSIFICATION FUNCTION COEFFICIENTS
(FISHER'S LINEAR DISCRIMINANT FUNCTIONS)

PATIENTS	1	2	3	4
1	-4.553044	-3.900489	-3.542848	-4.243995
2	1.173253	1.173253	1.055974	1.202208
3	1.173253	1.173253	1.055974	1.202208
4	1.173253	1.173253	1.055974	1.202208
5	1.173253	1.173253	1.055974	1.202208
6	1.173253	1.173253	1.055974	1.202208
7	1.173253	1.173253	1.055974	1.202208
8	1.173253	1.173253	1.055974	1.202208
9	1.173253	1.173253	1.055974	1.202208
10	1.173253	1.173253	1.055974	1.202208
11	1.173253	1.173253	1.055974	1.202208
12	1.173253	1.173253	1.055974	1.202208

CANONICAL DISCRIMINANT FUNCTIONS

FUNCTION	EIGENVALUE	PERCENT VARIANCE	CUMULATIVE PERCENT	CANONICAL CORRELATION	FUNCTION AFTER	WILKS' LAMBDA	CHI-SQUARED	D.F.	SIGNIFICANCE
1	5.20057	45.28	45.28	0.9440927	1	0.013041	72.807	14	0.0000
2	5.20057	45.28	90.56	0.8867162	1	0.013041	72.807	14	0.0000
3	5.20057	45.28	100.00	0.6369809	1	0.013041	72.807	14	0.0000

* MARKS THE 3 CANONICAL DISCRIMINANT FUNCTION(S) TO BE USED IN THE REMAINING ANALYSIS.

Figure 9. Summary Table of Discriminant Analysis on Tank

This means that a lambda of this magnitude or smaller has a .1823 probability of occurring due to the chances of sampling even if there was no further information to be accounted for by a third function in the population. Clearly, a third function is not statistically significant in this case.

The standardized discriminant function coefficients corresponding to the values of the v_{ij} 's discussed in the previous section are used to compute the discriminant score for a case (observation) in which the original discriminating variables are in standard form. The discriminant score is computed by multiplying each discriminating variable by its corresponding coefficient and adding together these products. There is a separate score for each observation on each function. The coefficients have been derived in such a way that the discriminant scores produced are in standard form.

When the sign is ignored, each standard discriminant function coefficient represents the relative contribution of its associated variable to that function. The sign merely denotes whether the variable is making a positive or negative contribution.

A graphical presentation is shown in Figure 11 using the first and the second canonical discriminant function as the axis. From this scatterplot, we can easily see that Soviet tanks (labelled 1) are well distinguished from the all of the others using only the first two discriminant

functions. Also, all the lightweight tanks are clearly separated from the others. The distinction between groups 2 and 3 is also clear though not separated from each other as much as from groups 1 and 4. The complete SPSS program for the discriminant analysis is listed in Appendix D.

VII. CONCLUSION

The multivariate analysis techniques of cluster analysis, principal components analysis and discriminant analysis are useful in real world problems for examining observations on each of several dimension. Each of the techniques is related mathematically to the others, and each complements the other in explaining the data.

Computer software is readily available in many sources. The software used in this thesis for hierarchical clustering, principal components analysis, and discriminant analysis was from the IMSL package and SPSS. For nonhierarchical clustering, we used the FORTRAN program developed by McRae (16). All of this software is readily available and documented at the Naval Postgraduate School.

HIERARCHICAL CLUSTERING

THIS IS A PROGRAM FOR HIERARCHICAL CLUSTERING

```

VARIABLE DESCRIPTION
ND      INPUT NUMBER OF DATA POINTS TO BE CLUSTERED.
        ND-1 CLUSTERS ARE FORMED NUMBERED CONSECUTIVELY
        FROM ND+1 TO ND+(NC-1). ND MUST BE GREATER THAN 2.
        ND IN THE RANGE 2,3,....,500 IS ALLOWED
        (SEE REMARKS)
INPUT OPTIONS VECTOR CF LENGTH 2
IOPT(1)=0 IMPLIES SINGLE-LINKAGE IS DESIRED.
        OTHERWISE, COMPLETE-LINKAGE IS DESIRED.
IOPT(2)=0 IMPLIES THE SIMILARITIES ARE DIRECTLY
        LIKE (I.E. SMALLER ARE ASSUMED TO BE POSITIVE
        THE SIMILARITIES ARE ASSUMED TO BE POSITIVE)
        CORRELATION-LIKE CF LENGTH ((NC+1)*NC)/2
INPUT/OUTPUT VECTOR CF LENGTH MATRIX IN SYMMETRIC
        CONTAINS MOCE. FOR J GREATER THAN J, SIMILARITY CF
        XSIM((I-1)*I+J) CONTAINS THE INPUT, THE
        THE I-TH AND J-TH DATA POINTS. CF ITSELF ARE
        DIAGONAL ELEMENTS (SIMILARITY CF ITSELF) ARE
        ARBITRARY AND DO NOT NEED TO BE DEFINED.
        XSIM IS DESTROYED ON OUTPUT.
CLEVEL  OUTPUT VECTOR CF LENGTH ND-1. CLEVEL(K) CONTAINS THE
        SIMILARITY LEVEL AT WHICH CLUSTER NC+K WAS FORMED.
DIMENSION XSIM(500), IOPT(2), CLEVEL(500), TITLE(20),
1ICRSCN(500), IPTR(100), IOUT(100), INCLST(100),
2LEFTRT(100), START(100), YSIM(2)
THIS WOULD BE THE INPUT VECTOR FOR SUBPROGRAM USTREE.
ICLSCN  OUTPUT VECTOR OF LEFTS OF LENGTH NC-1. CLUSTER
        NUMBER ND+K WAS FORMED BY MERGING CLUSTER ICLSCN(K)
        WITH CLUSTER ICRSCN(K). VECTOR FOR SUBPROGRAM USTREE.
        THIS WOULD BE THE INPUT VECTOR FOR SUBPROGRAM USTREE.
        OUTPUT VECTOR OF RIGHTS OF LENGTH NC-1. THE
        RIGHTSCN OF CLUSTER ND+K IS CONTAINED IN ICRSCN(K).
        THIS WOULD BE THE INPUT VECTOR FOR SUBPROGRAM USTREE.
        WORK VECTOR CF LENGTH ND.
        ERROR PARAMETER. (CUTPUT)
TERMINAL ERROR
IER=129 INDICATES ND WAS LESS THAN 3.
INPUT VECTOR OF LENGTH 4. IND(1) CONTAINS WHEN I=1,
        THE HEAD CODE OF THE SUBTREE TO BE PRINTED. (REMARKS)
        MUST BE BETWEEN NC AND 2*NC, EXCLUSIVELY. (REMARKS)
        I=2, NUMBER OF PRINTABLE SPACES PER LINE ON THE
        OUTPUT (PRINTER) DEVICE. IND(2) MUST EXCEED 4.

```



```

I=3, NUMBER OF HORIZONTAL SLICES OF TREE DESIRED
TO PROVIDE THE NECESSARY DETAIL.
INC(3) MUST BE POSITIVE.
I=4, NUMBER OF FILLER LINES PRINTED BETWEEN NODE
LINES (1 USUALLY SUFFICIENT).
INPUT VECTOR OF LENGTH 2 CONTAINING THE INTERVAL CN
THE VECTOR SCALE USED TO PLOT THE TREE
(SEE VECTOR CLEVEL). LEVEL YSIM(1) IS WHERE THE
TERMINAL NODES (FOR IMSLCLUSTERING ROUTINES, THE
DATA POINTS) ARE PRINTED. THE OTHER INTERVAL
ENCPOINT YSIM(2) SHOULD INCLUDE THE LEVEL FOR THE
FEAC NODE (INC(1)). SEE REMARKS.

YSIM
NAMELIST /NAME1/XSIM
ICUT WORK VECTOR OF LENGTH INC(2)-4.
CLVLSK WORK VECTOR OF LENGTH ND.
NCLRST WORK VECTOR OF LENGTH ND.
LEFTRT WORK VECTOR OF LENGTH ND.
STARST WORK VECTOR OF LENGTH ND.

REMARKS 1. THE DATA CLUSTERS ARE NUMBERED 1 TO NC. THE ND-1 CLUSTERS
FORMED BY MERGING ARE NUMBERED ND+1 TO NC+(ND-1) AND DECREASE
IN SIMILARITY, MAKING IT EASY TO IDENTIFY THE MOST SIMILAR
CLUSTERS.

2. SIMILARITIES GENERALLY SHOULD BE NONNEGATIVE. RAW
CORRELATIONS TAKE CN VALUES R WHERE R LIES IN THE CLOSED
INTERVAL (-1, 1) AND SHOULD BE MADE POSITIVE. IF R=1 AND
R=-1 BOTH MEAN HIGH SIMILARITY, THEN THE TRANSFORMATIONS, SQUARED,
RR EQUAL TO THE ABSOLUTE VALUE OF R, CF, RR EQUAL TO R SQUARED,
ARE APPROPRIATE. IF R=-1 REPRESENTS A VERY LOW SIMILARITY, THEN
THE TRANSFORMATION, RR=1-R, BECOMES A DISTANCE-LIKE SIMILARITY.

3. NOTE THAT THE ORIGINAL DATA MATRIX OF THE USER DOES NOT
ENTER CCLINK, ALLOWING THE USER TO DEFINE, VIA XSIM,
WHATEVER MEASURE OF SIMILARITY SEEMS MOST APPROPRIATE.

4. THE TREE MAY BE TOO LARGE TO FIT IN INC(2) SPACES
REPRESENTING THE INTERVAL (YSIM(1), YSIM(2)). IF SO, LSTREE CAN
PRINT THE TREE IN INC(3) SLICES OF THIS INTERVAL WHICH MAY BE
CUT AND TAPED TOGETHER.

READ(5, 11C) (TITLE(I), I=1, 20)
5. TO PRINT THE ENTIRE TREE FROM IMSL SUBROUTINE CCLINK, THE
HEAC NODE INC(1) = NC+(ND-1).

6. OUTPUT IS WRITTEN TO THE UNIT SPECIFIED BY IMSL ROUTINE
UGETIO. SEE THE UGETIO DOCUMENT FOR DETAILS.

```


7. REVERSALS (IER=130) MAY OCCUR IN TWO WAYS. FIRST, TWO NODES
MAY BE JOINED AT A LOWER LEVEL (CLOSER TO YSIM(1)). SECOND, THE
LEVEL OF THE HEAD NODE MAY LIE ABOVE THE INTERVAL
(YSIM(1), YSIM(2)).
8. FOR PROPER DISPLAY, THE TREE CREATED BY USTREE SHOULD BE
TURNED TO AN UPRIGHT POSITION.

WRITE(6,111) (TITLE(I),I=1,20)

READ THE NUMBER OF OBSERVATION AND
TWO INPUT OPTIONS.

READ(5,112) ND, IOPT(1), IOPT(2)

WRITE(6,200) ND

IF(NC.LE.2) GO TO 50

N=((ND+1)*NC)/2

IF(IOPT(1).EQ.0) GC TC 30

WRITE(6,113)

IF(IOPT(2).EQ.0) GC TC 31

WRITE(6,114)

GO TO 32

WRITE(6,115)

GO TO 29

WRITE(6,116)

WRITE(6,13) N

READ INPUT DATA MATRIX BY THE

NAME LIST FORMAT

AN ARRAY OF VECTOR FROM THE LOWER TRIANGULAR MATRIX.

READ (5,NAM1)

WRITE (6,300)

JJJJ=1

DO 10 I=1,NC

J-KK=JJJJ+I-1

WRITE (6,101) I, (XSIM(JK), JK=JJJJ, JKK)

JJJJ=JJJJ+1

WRITE (6,100) (I, I=1, ND)

CALL IMSL SUBPROGRAM OCLINK

CALL OCLINK(ND, ICFT, XSIM, CLEVEL, ICLSON, ICRSCN, IPTF, IER)

WRITE(6,1)

WRITE THE SIMILAFITY LEVEL, ICLSON AND
ICRSCN TO CONSTRUCT A TREE.


```

WRITE(6,2) (CLEVEL(I),ICLSON(I),ICRSON(I),I=1,25)
INC(1)=2*NC-1
INC(2)=70
INC(3)=1
INC(4)=1
YSIM(1)=CLEVEL(1)
YSIM(2)=0.0

      CALL INSL USTREE FOR THE GRAPHICAL
      PRESENTATION.

CALL USTREE(ND,ICLSON,ICRSON,CLEVEL,IND,YSIM,IOUT,C353SK,ACLRST,
1LEFTRT,STARST,IER)
GC TO GC
WRITE(6,5CC)
STOP
FCRMAT(1H1,5X,'CLEVEL',6X,'ICLSON',6X,'ICRSON')
FCRMAT(2X,F10.1,2X,I7,5X,I7)
FCRMAT(30,'NUMBER OF OBSERVATIONS IN XSIM ARE',I6)
FCRMAT(4,5X,I4)
FCRMAT(50,'I5,26F4.0)
FCRMAT(620A4)
FCRMAT(71,20A4)
FCRMAT(81,20A4)
FCRMAT(914,211)
FCRMAT(10COMPLETE-LINKAGE IS PERFORMED.')
FCRMAT(11SIMILARITIES ARE POSITIVE CORRELATION-LIKE.')
FCRMAT(12SIMILARITIES ARE POSITIVE CORRELATION-LIKE.')
FCRMAT(13SINGLE LINKAGE IS PERFORMED.')
FCRMAT(14SINGLE LINKAGE IS PERFORMED.')
FCRMAT(15SIMILARITIES ARE DISTANCE-LIKE.')
FCRMAT(16SIMILARITIES ARE DISTANCE-LIKE.')
FCRMAT(17NUMBER OF DATA POINTS ARE',I4)
FCRMAT(18NUMBER OF DATA POINTS ARE',I4)
FCRMAT(19INITIAL DISTANCE MATRIX',//)
FCRMAT(20YOUR DATA ARE TOO SMALL')
FCRMAT(21)
EN

```

CC

(HIERARCHICAL CLUSTERING AND UTILITY DATA)

NUMBER OF DATA POINTS ARE 34

SINGLE IMAGE IS FEEBLE.

SIMILARITIES ARE POSITIVE CORRELATION-LIKE.

NUMBER OF OBSERVATIONS IN Δ SIN ARE 351

INITIAL DISTANCE MATRIX

1	5C.																									
2	4.	5C.																								
3	8.	1.	5C.																							
4	7.	1.	11.	5C.																						
5	5	C.	C.	C.	C.	5C.																				
6	6	C.	C.	C.	C.	8.	5C.																			
7	7	C.	C.	C.	C.	8.	12.	50.																		
8	6	C.	1.	C.	C.	3.	3.	3.	5C.																	
9	5	C.	C.	C.	C.	1C.	1C.	3.	50.																	
10	1C	C.	C.	1.	1.	C.	C.	3.	C.	5C.																
11	11	C.	C.	C.	C.	6.	6.	2.	7.	C.	50.															
12	12	C.	C.	C.	C.	0.	1.	3.	3.	1.	2.	C.	5.	50.												
13	13	C.	C.	C.	C.	1.	3.	3.	1.	2.	C.	5.	12.	50.												
14	14	C.	C.	C.	C.	2.	5.	5.	2.	4.	C.	5.	9.	5.	5C.											
15	15	C.	C.	C.	C.	C.	C.	0.	3.	0.	5.	0.	0.	0.	C.	50.										
16	16	C.	C.	C.	C.	3.	5.	5.	0.	4.	1.	6.	3.	3.	3.	5C.										
17	17	C.	C.	C.	C.	0.	2.	1.	1.	2.	C.	5.	3.	3.	2.	0.	2.	5C.								
18	18	C.	C.	C.	C.	1.	1.	1.	1.	1.	C.	4.	3.	4.	2.	0.	1.	5.	5C.							
19	19	C.	C.	C.	C.	0.	5.	5.	5.	2.	7.	C.	3.	0.	C.	C.	0.	2.	2.	1.	50.					
20	20	2.	3.	1.	1.	5.	1.	1.	1.	3.	C.	2.	0.	C.	C.	0.	0.	2.	1.	8.	50.					
21	21	C.	C.	C.	C.	0.	1.	C.	C.	1.	C.	1.	0.	C.	C.	0.	0.	2.	C.	3.	5.	5C.				
22	22	2.	8.	C.	C.	C.	C.	0.	1.	0.	C.	0.	0.	C.	C.	0.	C.	C.	C.	2.	2.	5C.				
23	23	2.	7.	C.	C.	C.	C.	0.	1.	0.	C.	0.	0.	C.	0.	0.	0.	C.	0.	2.	3.	11.	5C.			
24	24	2.	6.	C.	C.	C.	C.	0.	1.	0.	C.	0.	0.	C.	C.	0.	C.	C.	0.	2.	3.	10.	11.	5C.		
25	25	2.	1.	C.	C.	C.	C.	0.	C.	0.	C.	0.	0.	C.	C.	0.	C.	C.	C.	2.	2.	12.	11.	10.	5C.	
26	26	C.	C.	C.	C.	0.	1.	C.	C.	0.	C.	0.	0.	C.	C.	0.	0.	1.	1.	2.	2.	5.	3.	4.	3.	5C.
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	

Appendix B:

K-MEANS ITERATIVE CLUSTERING PROGRAM BY D.J. MCRAE

THIS PROGRAM ENABLES THE USER TO CLUSTER A DATA MATRIX UP TO SIZE (600 X 20) INTO A MAXIMUM OF 20 CLUSTERS. SEVERAL OPTIONS ARE AVAILABLE WITH RESPECT TO THE METHOD OF COMPUTATION AND CRITERIA TO ACCOMPLISH THE CLUSTERING. THE PROGRAM MAY BE USED ON EITHER THE CP/CMS TERMINALS OR BY THE CARD READER. THE APPROPRIATE METHOD FOR EACH IS DESCRIBED BELOW.

CP/CMS TERMINAL USE
THE PROGRAM SHOULD BE READ INTO CP AS IS, BUT PRECEDED BY THE FOLLOWING CARDS, STARTING IN COLUMN ONE

CF67USERID XXXXG
CFFLINE READ RRREAD FCRTAN

THE DATA MATRIX CAN BE PLACED IN THE COMPILE EITHER BY USE OF THE CFFLINE READ OR BY TYPING IN THE INFORMATION ON THE TERMINAL. IF CFFLINE READ IS USED THE DATA MUST BE IN THE SPECIFIED FORMAT PRECEDED BY THE FOLLOWING TWO CARDS, STARTING IN COLUMN ONE:

CF67USERID XXXXG
CFFLINE READ FILE FT04FYVY

THE PROGRAM CAN THEN BE EXECUTED BY FIRST ISSUING THE

CMS COMMAND
FOLLOWED BY

F RRREAD
\$ RRREAD

CS/CARD READER USE

THE PROGRAM MUST CONTAIN THE FOLLOWING INFORMATION IN THE FOLLOWING ORDER:

STANDARD GREEN JOB CARD, TIME=(AS DESIRED)
// EXEC FORTCLG, RECIEN, GO=20CK
// FCRT, SYSIN DD *
// MAIN PROGRAM
/*
//GO, SYSIN DD *
//GO, DATA DECK
/*

RRRCC120
RRR00010
RRR00020
RRRCC030
RRR00040
RRR00050
RRR00060
RRR00070
RRR00080
RRR00090
RRRCC100
RRRCC110
RRR00120
RRR00130
RRR00140
RRRCC150
RRR00160
RRRCC170
RRR00180
RRRCC190
RRR00200
RRRCC210
RRR00220
RRR00240

RRR00270
RRR00280

RRR00330
RRR00340

RRR00360
RRRCC370

RRRCC400
RRR00410
RRRCC420
RRR00430
RRR00440
RRR00450
RRR00460
RRR00470
RRR00480

DATA INPUT DECK

THE FIRST CARD OF THE DATA DECK IS THE TITLE CARD.
IT MAY CONTAIN ANY ALPHA NUMERIC TITLE IN COLUMNS 1 - 80.

THE SECOND CARD IS THE PROBLEM CARD. IT CONTAINS
INTEGERS IN THE FIRST 13 COLUMNS IN THE FOLLOWING MANNER:

CCL 1-4: NUMBER OF OBSERVATIONS
CCL 5-6: NUMBER OF VARIABLES PER OBSERVATION
CCL 7-8: NUMBER OF CLUSTERS DESIRED
CCL 9: CRITERION TO BE USED IN THE EVALUATION:

1 = TRACE W
2 = DETERMINANT W
3 = LARGEST EIGENVALUE OF W (INVERSE)*B
4 = TRACE W (INVERSE)*B

CCL 10: STANDARDIZATION PARAMETER:
C = THE PROGRAM WILL NOT STANDARDIZE DATA
1 = THE PROGRAM WILL STANDARDIZE DATA

CCL 11: DISTANCE PARAMETER
0 = EUCLIDIAN DISTANCE
1 = SCALED EUCLIDIAN DISTANCE
2 = MAHALANOBIS DISTANCE

CCL 12: DATA PARAMETER
C = DATA IS TO BE READ IN
1 = NO DATA REANALYZED USING CURRENT PROBLEM

CCL 13: TIMING PARAMETER
0 = NO OBSERVATIONS CONSIDERED IN
INDIVIDUAL SWITCHES
1 - 8 = AN INCREASING NUMBER OF OBSERVATIONS
9 = ALL OBSERVATIONS USED
NOTE: THE TIMING PARAMETER DETERMINES HOW EXTENSIVE THE
DOUBLE CHECK OF THE CLUSTER SOLUTION IS TO BE.
NORMALLY '9' IS USED UNLESS THE DATA MATRIX IS LARGE

THIS CARD MUST BE IN (14,212,511) FORMAT

THE THIRD CARD IS THE VARIABLE FORMAT CARD. THIS CARD
MUST BE PRESENT UNLESS THE DATA PARAMETER IN THE PROBLEM
CARD IS '1'. THE FORMAT STATEMENT IS THAT TYPE WHICH THE DATA
CARDS ARE IN. FOR EXAMPLE IF THE DATA CARD HAS DATA:
ABCD 1.55
STARTING IN COL 1, THE VARIABLE FORMAT CARD WOULD
CONTAIN THE FOLLOWING IN CCL 1:
(1A4,4X,F3.1,4X,F4.2)

THE NEXT CARDS ARE THE DATA CARDS. THEY MAY CONTAIN
DATA IN COLUMNS 1 - 72.

RRR000490
RRR000500
RRR000510
RRR000520
RRR000530
RRR000540
RRR000550
RRR000560
RRR000570
RRR000580
RRR000590
RRR000600
RRR000610
RRR000620
RRR000630
RRR000640
RRR000650
RRR000660
RRR000670
RRR000680
RRR000690
RRR000700
RRR000710
RRR000720
RRR000730
RRR000740
RRR000750
RRR000760
RRR000770
RRR000780
RRR000790
RRR000800
RRR000810
RRR000820
RRR000830
RRR000840
RRR000850
RRR000860
RRR000870
RRR000880
RRR000890
RRR000900
RRR000910
RRR000920
RRR000930
RRR000940
RRR000950


```

      FCR REANALYSIS OF THE SAME DATA, ACC A NEW TITLE CARD
      FOLLOWED BY A NEW PROBLEM CARD, WITH A '1' IN COLUMN 11
      CF THE NEW PROBLEM CARD.
      FCR ANALYSIS OF NEW DATA, SIMPLY PLACE THE NEW DATA
      SET, INCLUDING NEW TITLE CARD, PROBLEM CARD, AND FORMAT CARD,
      AFTER THE PREVIOUS SET
      FCR THE PROGRAM TO EXIT NORMALLY, TWO BLANK CARDS MUST FOLLOW
      THE LAST DATA BATCH.
      CCMCN NOBS, NVAR, NGPS, ICRIT, NOSTAN, IDIST, IFINE, KTIME, IDENT(600),
      1 IDATA(600), NISV(20), VMEAN(20), SC(20), XVEC(20), YVEC(20), BT(20,20),
      2 IDATA(600), NISV(20), SVCENT(20,20), IDATAT(600), VEC(20,20), EIG(20)
      3 NISV(20), SVCENT(20,20), IDATAT(600), VEC(20,20), EIG(20)

      DESCRIPTION OF COMMON AREA:
      NOBS = NUMBER OF OBSERVATIONS
      NVAR = NUMBER OF VARIABLES
      NGPS = NUMBER OF CLUSTERS
      ICRIT = CRITERION TO BE OPTIMIZED (SEE ABOVE)
      NCSTAN = STANCE PARAMETER (SEE ABOVE)
      IDIST = DISTANCE PARAMETER (SEE ABOVE)
      IFINE = ESCAPE PARAMETER: IF IFINE ABOVE IS SET EQUAL TO '1',
      SOMETHING IS WRONG AND THE APPROPRIATE ERROR
      MESSAGE IS PRINTED OUT; THE PROGRAM GOES ON TO THE
      NEXT PROBLEM MEETER (SEE ABOVE)
      KTIME = TIMING PARAMETER (SEE ABOVE)
      IDENT = OBSERVATION IDENTIFICATIONS (A4 FORMAT): READ FROM
      EACH DATA CARD
      DATA = AREA IN WHICH THE DATA MATRIX IS STORED
      T = CROSS-PRODUCTS MATRIX
      B = BETWEEN-CLUSTERS MATRIX
      W = WITHIN-CLUSTERS MATRIX
      WFACT = CHCLESKY FACTOR OF THE WITHIN-CLUSTERS MATRIX
      SVCENT = CLUSTER CENTERS (MEANS)
      IDATA = CLUSTER IDENTIFICATION FOR EACH OBSERVATION
      NISV = CLUSTER SIZES (NUMBER OF OBSERVATIONS IN EACH CLUSTER)
      VMEAN = VARIABLE MEANS
      SD = VARIABLE STANDARD DEVIATIONS
      XVEC = TEMPORARY STORAGE
      YVEC = TEMPORARY STORAGE
      NNN=24
      BT = TEMPORARY STORAGE FOR BETWEEN-CLUSTERS MATRIX
      NISV, SVCENT, IDATAT: TEMPORARY STORAGE SERVING THE
      SAME FUNCTIONS AS NISV, SVCENT, AND IDATA

```

```

RRRC0570
RRRC0580
RRRC0590
RRR01000
RRR01010
RRR01020
RRR01030
RRR01040
RRR01050
RRR01060
RRR01070
RRR01080
RRR01090
RRR01100
RRR01110
RRR01120
RRR01130
RRR01140
RRR01150
RRR01160
RRR01170
RRR01180
RRR01190
RRR01200
RRR01210
RRR01220
RRR01230
RRR01240
RRR01250
RRR01260
RRR01270
RRR01280
RRR01290
RRR01300
RRR01310
RRR01320
RRR01330
RRR01340
RRR01350
RRR01360
RRR01370
RRR01380
RRR01390
RRR01400
RRR01410
RRR01420
RRR01430

```

```

CCCCCCCCCCCC
CCCCCCCCCCCC
CCCCCCCCCCCC
CCCC

```



```

C          VEC = EIGENVECTORS
C          FIG = EIGENVALUES
C
1CC      CALL PRELIM (CRIT)
C      IFINE IS ESCAPE PARAMETER
C      IF (IFINE.EQ.1) GC TO 400
C      IF (IFINE.EQ.2) GC TO 500
C      CALL IRANDST (CRIT)
C      IF (IFINE.EQ.1) GC TO 400
3CC      CALL KMEANS (CRIT)
C      IF (IFINE.EQ.1) GC TO 400
C      CALL ISWITCH (CRIT)
C      GC TO 100
C      IFINE GOT SET TO 1: IF DATA HAS BEEN READ IN, RESET DATA AND
C      GC TO NEXT PROBLEM
C      GC 410 I=1, NOBS
C      DC 410 J=1, NVARS
4CC      IF (NCSTAN.EQ.1) DATA(I,J) = DATA(I,J) * SD(J)
C      DATA(I,J) = DATA(I,J) + VMEAN(J)
C      GC TO 100
C      ALL DCNE: WRITE CUT LAST MESSAGE AND EXIT
5CC      WRITE (6,50C)
C      FCRMAT (10 END OF ANALYSES.)
C      STOP
C      ENCL
C      SUBROUTINE FRELIM (CRIT)
C
C      THIS SUBROUTINE MAKES THE PRELIMINARY CALCULATIONS.
C      IT INPUTS THE DATA, CALCULATES THE MEANS AND VARIANCES FOR EACH
C      VARIABLE, STANDARDIZES (CONVERTS TO Z-SCORES) EACH VARIABLE IF
C      REQUESTED, AND CALCULATES THE CROSS-PRODUCTS MATRIX
C
C      COMMON NOBS, NVARS, NGPS, ICRT, NOSTAN, IDIST, IFINE, KTIME, IDENT(6C0),
1      IDATA(6C0,20), T(20,20), B(20,20), W(20,20), WFC1(20,20), SVCEN(2C,2C),
2      IDATA(6C0), NISV(2C), VMEAN(20), SC(2C), XVEC(2C), YVEC(20), ET(20,20),
3      NISVT(20), SVCENT(20,20), IDATAT(6C0), VEC(20,20), EIG(2C)
C      DIMENSION TITLE(20), IFMT(20), VAR(2C)
C
C      INPUT SECTION: READ IN TITLE CARD, PROBLEM CARD, OPTIONAL CARDS,
C      FCRMAT CARD, AND DATA CARDS
C      WRITE OUT SOLUTION SPECIFICATIONS
C
C      IFINE = 0
C      REAC (5,900) (TITLE(1), I=1,20)
C      WRITE (6,503) (TITLE(I), I=1,20)
C      READ (5,501) NOBS, NVARS, NGPS, ICRT, NOSTAN, IDIST, NCCDATA, KTIME
C      IF (NOBS.EQ.0) GC TO 820

```

```

RRR01440
RRR0146C
RRR01470
RRRC148C
RRR01490
RRR0150C
RRR01510
RRR01520
RRR01530
RRR01540
RRR01550
RRR01560
RRRC157C
RRR01580
RRRC159C
RRR01600
RRRC161C
RRR0163C
RRR01640
RRR0165C
RRR01660
RRRC1670
RRR0168C
RRRC169C
RRRC170C
RRR01710
RRR01720
RRR01730
RRRC1740
RRR01750
RRR0176C
RRR01770
RRRC1780
RRR01790
RRR01800
RRRC181C
RRR01820
RRR01830
RRRC1840
RRR01850
RRRC186C
RRR0187C
RRR01880
RRR01890
RRR01900
RRR01910

```



```

700 CCATINUE ('C',10(F11.3,1X))
701 FCRMAT ('1',,STANDARDIZED DATA MATRIX IS')
702 WRITE (6,912)
      DC 170 I=1,NVARS
      WRITE (6,701) (T(J,I),J=1,I)
17C CCATINUE
912 FCRMAT('O THE STANDARDIZED CROSS-PRCD MATRIX IS ')
18C RETURN = 1
815 IF (NODATA.EQ.0) IFINE=2
      WRITE (6,935)
      RETURN = 2
820 IFINE = 2
825 RETURN
      WRITE (6,945) J
      IFINE = 1
      RETURN

CC C
FCRMAT STATEMENTS
900 FCRMAT (20A4)
901 FCRMAT (14,212,511)
902 FCRMAT (18A4)
903 FCRMAT ('1',20A4)
904 FCRMAT ('O THE NUMBER OF OBSERVATIONS IS ',14,/, ' THE NUMBER OF VARIAB
1ARIABLES IS ',12,/, ' THE INPUT FORMAT IS ',18A4)
905 FCRMAT ('O',5(F11.3,1X),/,1X,5(F11.3,1X),//)
907 FCRMAT ('O',, ' THE VARIABLE MEANS ARE ')
908 FCRMAT ('1',, ' THE VARIABLE STANDARD DEVIATIONS ARE ')
909 FCRMAT ('O THE CROSS-PRODUCTS MATRIX IS ')
910 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE W')
920 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE LARGEST EIGENVALUE W')
921 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
922 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
923 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
924 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
925 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
926 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
927 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
928 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
929 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
930 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
931 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
932 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
933 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
934 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
935 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
936 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
937 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
938 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
939 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
940 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
941 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
942 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
943 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
944 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
945 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
946 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
947 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
948 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
949 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
950 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
951 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
952 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
953 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
954 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
955 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
956 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
957 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
958 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
959 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
960 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
961 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
962 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
963 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
964 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
965 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
966 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
967 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
968 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
969 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
970 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
971 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
972 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
973 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
974 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
975 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
976 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
977 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
978 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
979 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
980 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
981 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
982 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
983 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
984 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
985 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
986 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
987 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
988 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
989 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
990 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
991 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
992 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
993 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
994 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
995 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
996 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
997 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
998 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')
999 FCRMAT ('O THE CRITERION IS TO MINIMIZE THE TRACE OF W-1E')

```



```

END
SLBRROUTINE RANDST (CRIT)
THIS SLBRROUTINE DETERMINES THE INITIAL CLUSTER CONFIGURATION.
COMMON NOES, NVAR, NGPS, ICRT, NOSTAN, IDIST, IFINE, KTIME, ICENT(6CC),
IDATA(6CC,20), T(20,20), B(20,20), W(20,20), WFC(20,20), SVCEN(20,20),
IICATA(6CC), NISV(20), VMEAN(20), SC(20), XVEC(20), YVEC(20), ET(20,20),
3 NISVT(20), SVCENT(20,20), IDATA(6CC), VEC(20,20), EIG(20)
DIMENSION ISTR(20)
CNE PASS K-MEANS TO DETERMINE INITIAL CLUSTER SOLUTION
RANDCN NUMBER STARTER
IX=19735
NGPST=NGPS
GO 260: GETTING THREE INITIAL CONFIGURATIONS; THE CNE WITH THE
BEST CRITERION VALUE WILL BE USED
DO 260 IBSRT = 1,3
DO 205 M = 1,NGPS
NISVT(M)=1
FINC A RANDOM OBSERVATION TO SERVE AS INITIAL CLUSTER CENTER
FOR EACH CLUSTER
DO 201 ISTR(M)=URANC(IX)
RANC = RAND * NOBS
IF (ISTR(M).EQ.0) ISTR(M)=1
IF (M.EC.1) GO TO 203
MM1 = M-1
DO 204 MM=1,MM1
IF (ISTR(M).EQ.ISTR(MM)) GO TO 201
CONTINUE
DO 203 I = ISTR(M)
DO 205 J=1, NVAR
SVCENT(M,J) = DATA(I,J)
ITEMP = IDIST
ICIST = 0
DO FOR EACH OBSERVATION
DO 225 I=1,NOBS
FINC EUCLIDIAN DISTANCE TO EACH CLUSTER CENTER
DO 215 M=1,NGPS
DO 210 J=1, NVAR
XVEC(J)=DATA(I,J)
YVEC(J)=SVCENT(M,J)
CALL DISTCE (XVEC,YVEC,DISTA)
IF (IFINE.EC.1) RETURN
IF (M.EC.1) GO TO 212
IF (DISTA.GE.SMDIST) GO TO 215
SMDIST = DISTA

```



```

215 ICATAT(I) = M
C CONTINUE
C ASSIGN OBSERVATION TO CLOSEST CLUSTER AND UPDATE THAT CLUSTER
C CENTER
K = IDATAT(I)
NISVT(K) = NISVT(K) + 1
CC 225 J = 1, NVARS
225 SVCENT(K,J) = ((NISVT(K)-1)*SVCENT(K,J)+DATA(I,J))/NISVT(K)
C ICIST = ITEMP
C ENCL CF DC FOR EACH CBSERVATION
C RECALCULATE CLUSTER CENTERS TO ELIMINATE INITIAL RANCCM CBSERVA-
C TIONS
CC 230 M=1, NGPS
NISVT(M) = NISVT(M)-1
CC 230 J=1, NVARS
230 SVCENT(M,J)=0.
CC 235 I=1, NOBS
M = IDATAT(I)
CC 235 J=1, NVARS
235 SVCENT(M,J) = SVCENT(M,J) + DATA(I,J)/NISVT(M)
C CALCULATE THE B AND W MATRICES
C CALCULATE THE CRITERION VALUE
ICIR=2
IF (ICRIT.EC.1) ICIR=1
CALL WCALC (SVCENT,NISVT,NGPST,IDIR)
IF (IFINE.EC.1) RETURN
CALL CRITON (CRIT) RETURN
IF (IFINE.EC.1) RETURN
WHICH INITIAL CONFIGURATION
IF (IBSRT.GT.1) GC TO 250
C FIRST INITIAL CONFIGURATION: BCRIT IS THE BEST CRITERION
245 BCRIT = CRIT
C SAVE CLUSTSV, AND OBSERVATION ID'S (ICATA)
LISTS (LSTSV), AND OBSERVATION ID'S (ICATA)
CC 247 M = 1, NGPS
NISVT(M) = NISVT(M)
CC 247 L = 1, NVARS
247 SVCENT(M,L) = SVCENT(M,L)
CC 249 I=1, NOBS
245 ICATA(I) = IDATAT(I)
CC TO 260
C SECOND OR THIRD INITIAL CONFIGURATION
250 IF (CRIT.GE.BCRIT) GO TO 260
255 BCRIT = CRIT
CC 257 M = 1, NGPS
NISVT(M) = NISVT(M)

```

```

RRRC368C
RRR03690
RRR03700
RRR03710
RRR03720
RRR03730
RRR03740
RRR03750
RRR03760
RRR0377C
RRR03780
RRR03790
RRR0380C
RRR03810
RRR03820
RRR03830
RRR03840
RRR03850
RRR0386C
RRR0387C
RRR0388C
RRR03890
RRR03900
RRR0391C
RRR03920
RRR03930
RRR03940
RRR03950
RRR03960
RRR0397C
RRR03980
RRR0399C
RRR04000
RRR04010
RRR0402C
RRR04030
RRR0404C
RRR04050
RRR0406C
RRR04070
RRR04080
RRR04090
RRR0410C
RRR04110
RRR04120
RRR0413C
RRR04140
RRR0415C

```



```

C 257 L = 1,NVARS
C 258 SVCEN(M,L) = SVCEN(M,L)
C 259 I=1,NCBS
C 260 ICAT(I) = ICAT(I)
C 261 CCENTINUE
C 262 ALL CCNE
C 263 RETURN
C 264 ENCL
C 265 SLROUTINE KMEANS (CRIT)
C 266
C 267 K-MEANS. EACH OBSERVATION IS ASSIGNED TO THE CLOSEST CLUSTER
C 268 CENTER.
C 269
C 270 CCMCN NCBS, NVARS, NGPS, ICRT, NOSTAN, IDIST, IFINE, KTIME, IDENT(600),
C 271 IDATA(600,20), T(20,20), R(20,20), W(20,20), WFC(20,20), SVCEN(20,20),
C 272 ICAT(600), NISV(20), VMEAN(20), SD(20), XVEC(20), YVEC(20), B1(20,20),
C 273 NISVT(20), SVCEN(20,20), IDATAT(600), VEC(20,20), EIG(20)
C 274 INITIALIZING FLAGS
C 275 ANL TEMPORARY STORAGE FOR NISV, SVCEN
C 276 NGPST = NGPS
C 277 JFLAG = CRIT
C 278 ECRT = CRIT
C 279 CC 201 M=1,NGPS
C 280 NISVT(M) = NISV(M)
C 281 CC 201 J=1, NVARS
C 282 SVCEN(M,J) = SVCEN(M,J)
C 283 RECALCULATING WITHIN-CLUSTERS MATRIX
C 284 ICIR=1
C 285 IF (IDIST.EC.2) ICIR=2
C 286 CALL WCALC (SVCEN,NISVT,NGPST,ICIR)
C 287 IF (IFINE.EC.1) RETURN
C 288 INITIALIZING TEMPORARY STORAGE FOR LISTV, IDATA
C 289 CC 210 I = 1,NOBS
C 290 ICAT(I) = ICAT(I)
C 291 CC 210 I = 1,NOBS
C 292 MAJOR DO LOOP: DO FOR EACH OBSERVATION
C 293 CC 400 I = 1,NOBS
C 294
C 295 BEGIN K-MEANS SECTION
C 296
C 297 CALCULATE VECTOR OF DISTANCES FROM OBSERVATION TO EACH CLUSTER
C 298 CENTER
C 299 CC 335 M=1,NGPST
C 300 CC 330 J=1, NVARS
C 301 XVEC(J) = ICAT(I,J)
C 302 YVEC(J) = SVCEN(M,J)
C 303 CALL DISTCE (XVEC,YVEC,DISTB)
C 304 IF (IFINE.EC.1) RETURN

```

```

RRRC4160
RRRC4170
RRRC4180
RRRC4190
RRRC4200
RRRC4210
RRRC4220
RRRC4230
RRRC4240
RRRC4250
RRRC4260
RRRC4270
RRRC4280
RRRC4290
RRRC4300
RRRC4310
RRRC4320
RRRC4330
RRRC4340
RRRC4350
RRRC4360
RRRC4370
RRRC4380
RRRC4390
RRRC4400
RRRC4410
RRRC4420
RRRC4430
RRRC4440
RRRC4450
RRRC4460
RRRC4470
RRRC4480
RRRC4490
RRRC4500
RRRC4510
RRRC4520
RRRC4530
RRRC4540
RRRC4550
RRRC4560
RRRC4570
RRRC4580
RRRC4590
RRRC4600
RRRC4610
RRRC4620
RRRC4630

```



```

IF (M.EQ.1) GO TO 332
IF (DISTR.GE.SMDIST) GO TO 335
IWDIST = DISTR
ICDGP = M
332 CCNTINUE
335 IS CBSERVATION ALREADY IN THE CLOSEST CLUSTER? IF YES, SKIP THIS
SECTION
IF (ICGP.EQ.IDATAT(I)) GO TO 360
ICLD IS OLD CLUSTER ASSIGNMENT
346 ICLC = IDATAT(I)
INEW IS NEW CLUSTER ASSIGNMENT
INEW = IDGP
JFLAG = 1
RECALCULATE CLUSTER CENTERS
DC 348 J = 1, NVAR$
SVCENT(IOLD,J) = (NISVT(IOLD)*SVCENT(IOLD,J)-DATA(I,J))/
1 (NISVT(IOLD)-1)
348 SVCENT(INEW,J) = (NISVT(INEW)*SVCENT(INEW,J)+CATA(I,J))/
1 (NISVT(INEW)+1)
ADJUST NISV
NISVT(IOLD) = NISVT(IOLD)-1
NISVT(INEW) = NISVT(INEW)+1
ADJUST ICATA
ICATAT(I) = IDGP
RECALCULATE WITHIN-CLUSTERS MATRIX FOR USE IN COMPUTING SCALED
EUCLIDIAN AND MAHALANOBIS DISTANCE
IF (IDIST.EC.0) GO TO 360
CALL WCALC (SVCENT,NISVT,NGPST,ICIR)
IF (IFINE.EC.1) RETURN
360 CCNTINUE
DC ONE WITH MAJOR DO LOOP
400 CCNTINUE

CALCULATE THE CRITERION
RECALCULATE SVCENT: ACCURACY MEASURE
DC 401 M=1, NGPST
DC 401 J=1, NVAR$
401 SVCENT(M,J) = 0.
DC 402 I=1, NCBS
M = IDATAT(I)
DC 402 J=1, NVAR$
SVCENT(M,J) = SVCENT(M,J) + CATA(I,J)/NISVT(M)
RECALCULATE WITHIN-CLUSTERS MATRIX AND CRITERION VALUE BASED ON
NEW CLUSTER CENTER VALUES
ICIR = 2
IF (ICRIT.EC.1) ICIR = 1
CALL WCALC (SVCENT,NISVT,NGPST,ICIR)

```



```

405 IF (IFINE.EC.1) RETURN
C CALL CRITCN (CRIT)
C IF (IFINE.EC.1) RETURN
C IF CRITERION BETTER THAN BEFORE? IF YES, THEN ANOTHER ITERATION:
C IF NO, FINISH KMEANS
C IF (CRIT.GE.BCRIT) GO TO 535
C
C ANOTHER ITEFATION
C
C PLT TEMPORARY VALUES INTO PERMANENT LOCATIONS
515 NGPS=NGPST
DC 520 M = 1,NGPS
NISV(M) = NISV(M)
DC 520 J = 1,NVARS
SVCEN (M,J) = SVCENT (M,J)
520 DC 530 I = 1,NOBS
530 ICATA(I) = IDATA(I)
GC TO 200
C
C FINISH
C
JFLAG = C MEANS NO CHANGES HAVE BEEN MADE DURING THE LAST
ITERATION: ITERATION IS CONVERGED
535 IF (JFLAG.EC.0) RETURN
JFLAG = 1 MEANS CHANGES HAVE BEEN MADE BUT THE CRITERION VALUE
GCT WCRSE: ITERATION IS NOT CONVERGING
540 WRITE (6,940) CRIT
WRITE (6,942) BCRIT
RECALCULATE WITHIN-CLUSTERS MATRIX AND RESET CRIT
ICIR=2
IF (ICRIT.EC.1) ICIR=1
CALL WCALC (SVCEN,NISV,NGPS,IDIR)
C IF (IFINE.EC.1) RETURN
C CRIT=BCRIT
C RETURN
540 FCRMAT ('O ITERATION IS NOT CONVERGING')
942 FCRMAT ('O THE CRITERION VALUE IS ',E12.6)
943 FCRMAT ('O THE BEST CRITERION VALUE IS ',E12.6)
END
SUBROUTINE ISWCH (CRIT)
C
C THIS SUBROUTINE CONSIDERS SWITCHING EACH OBSERVATION TO A
C DIFFERENT CLUSTER. THE SWITCH IS MADE IFF A BETTER CRITERION
C VALUE RESULTS.
C THIS SUBROUTINE ALSO DEPENDS ON THE PARAMETER KTIME: IT DETERMINES
C WHICH OBSERVATIONS ARE TO BE CONSIDERED. FOR COMPLETE EXPLANATION
C OF THE KTIME PARAMETER, SEE THE PROGRAM DESCRIPTION.

```



```

C      CCMCN NOBS, NVAR, NGPS, ICRIT, NOSTAN, IDIST, IFINE, KTIME, IDENT(6CC),
1 DATA(6CC,20), T(20,20), B(20,20), WFC1(20,20), SVCEN(20,20),
2 ICAT(6CC), NISV(20), VMEAN(20), SC(20), XVEC(20), YVEC(20), ET(20,20),
3 NISVT(20), SVCENT(20,20), IDATAT(6CC), VEC(20,20), EIG(20)
C      DIMENSION JFLAG(20)
C      SET UP THE ICR FLAG
C      ICR=2
C      IF (ICRIT.EC.1) ICR=1
C      KTIME = 0 MEANS SKIP THIS HEURISTIC
C      IF (KTIME.EC.0) GO TO 750
C      SET UP TEMPORARY STORAGE AREAS FOR NGPS, NISV, AND SVCEN
C      NGPS=NGPS
C      DC 500 M=1,NGPS
C      NISVT(M)=NISV(M)
C      JFLAG(M)=0
C      DC 500 J=1,NVAR
C      SVCENT(M,J)=SVCEN(M,J)
500 SET IFLAG = C: WILL BE SET = 1 IF ANY SWITCH IS MADE
600 IFLAG=0
C      MAJOR DO LGOF: DC 700
C      DC 700 M=1,NGPS
C      JFLAG(M)=1 INDICATES WHETHER CLUSTER M HAS BEEN ALTERED
C      IF (JFLAG(M).EQ.2) GO TO 700
C      KTIME = 0 MEANS ALL OBSERVATIONS WILL BE CONSIDERED
C      IF (KTIME.EC.9) GO TO 617
C      COMPUTE ALL DISTANCES FROM OBSERVATIONS TO CLUSTER CENTERS FOR THE
C      OBSERVATIONS IN CLUSTER M
C      DC 605 J=1,NVAR
C      YVEC(J) = SVCEN(M,J)
605 BIGDIS = 0
C      DC 615 I=1,NOBS
C      IF (IDAT(I).NE.M) GO TO 615
C      DC 610 J=1,NVAR
C      XVEC(J) = DATA(I,J)
610 CALL DISTCE(XVEC,YVEC,DISTA)
C      LOCATE OBSERVATION WITH BIGGEST DISTANCE
C      IF (BIGDIS.GE.DISTA) GO TO 615
C      EIGDIS = DISTA
615 CCNTINUE
C      CLUSTER BIGDIS BY TIMING PARAMETER KTIME
C      KTIME = KTIME
C      PSME = BIGDIS / TIME
617 MCLOD = M
C      DC FOR ALL OBSERVATIONS IN CLUSTER M
C      DC 691 I=1,NOBS
C      IF (IDAT(I).NE.M) GO TO 691
C      KFLAG = 0

```

```

RRRC5600
RRRC5610
RRRC5620
RRRC5630
RRRC5640
RRRC5650
RRRC5660
RRRC5670
RRRC5680
RRRC5690
RRRC5700
RRRC5710
RRRC5720
RRRC5730
RRRC5740
RRRC5750
RRRC5760
RRRC5770
RRRC5780
RRRC5790
RRRC5800
RRRC5810
RRRC5820
RRRC5830
RRRC5840
RRRC5850
RRRC5860
RRRC5870
RRRC5880
RRRC5890
RRRC5900
RRRC5910
RRRC5920
RRRC5930
RRRC5940
RRRC5950
RRRC5960
RRRC5970
RRRC5980
RRRC5990
RRRC6000
RRRC6010
RRRC6020
RRRC6030
RRRC6040
RRRC6050
RRRC6060
RRRC6070

```



```

C      KTIME = 9 MEANS CCNSIDER ALL OBSERVATIONS
C      IF (KTIME.EQ.9) GO TO 618
C      CCNSIDER SWITCHING THE OBSERVATION IF THE DISTANCE TC ITS
C      CURRENT CLUSTER CENTER IS LT BIGGIS / KTIME
C      CC 619 J=1,NVARS
C      615 XVEC(J) = DATA(I,J)
C      CALL DISTCE (XVEC,YVEC,DISTA)
C      IF (DISTA.LT.PSME) GO TO 691
C      CC CONTINUE
C      CC 690 MNEW=1,NGFS
C      IF (NISV(MOLD).EQ.1) GO TO 690
C      IF (MNEW.EQ.MOLD) GO TO 690
C      IF (KFLAG.EQ.1) GO TO 690
C      CC COMPUTE NEW SVCEN BASED ON OBSERVATION BEING SWITCHED
C      CC 620 J=1,NVARS
C      SVCENT(MNEW,J) = (NISV(MNEW)*SVCEN(MNEW,J)+DATA(I,J))/
C      1 (NISV(MNEW)+1)
C      1 SVCENT(MOLD,J) = (NISV(MOLD)*SVCEN(MOLD,J)-DATA(I,J))/
C      1 (NISV(MOLD)-1)
C      CC CONTINUE
C      62C ADJUST NISV
C      NISVT(MNEW) = NISV(MNEW)+1
C      NISVT(MOLD) = NISV(MOLD)-1
C      CC COMPUTE WITH IN-CLUSTERS MATRIX AND NEW CRITERION VALUE
C      CALL WCALC (SVCENT,NISVT,NGPST,ICIR)
C      IF ((IFINE.EQ.1) RETURN
C      CALL CRITON (TCRIT)
C      IF ((IFINE.EQ.1) RETURN
C      MAKE THE SWITCH IFF NEW CRIT IS BETTER
C      IF (TCRIT.GE.CRIT) GO TO 680
C      MAKE THE SWITCH
C      RESET FLAGS
C      JFLAG(M) = 1
C      JFLAG(MNEW) = 1
C      IFLAG=1
C      KFLAG = 1
C      ADJUST CRIT, IDATA, NISV, SVCEN
C      CRIT=TCRIT
C      ICATA(I) = MNEW
C      NISV(MNEW) = NISVT(MNEW)
C      NISV(MOLD) = NISVT(MOLD)
C      625 CC 630 J=1,NVARS
C      SVCENT(MNEW,J) = SVCENT(MNEW,J)
C      63C SVCEN(MOLD,J) = SVCENT(MOLD,J)
C      GO TO 690
C      SWITCH WAS NOT MADE; RESET NISVT, SVCENT, TC VALUES PFRESENT

```

```

RRRC6080
RRRC6090
RRRC6100
RRRC6110
RRRC6120
RRRC6130
RRRC6140
RRRC6150
RRRC6160
RRRC6170
RRRC6180
RRRC6190
RRRC6200
RRRC6210
RRRC6220
RRRC6230
RRRC6240
RRRC6250
RRRC6260
RRRC6270
RRRC6280
RRRC6290
RRRC6300
RRRC6310
RRRC6320
RRRC6330
RRRC6340
RRRC6350
RRRC6360
RRRC6370
RRRC6380
RRRC6390
RRRC6400
RRRC6410
RRRC6420
RRRC6430
RRRC6440
RRRC6450
RRRC6460
RRRC6470
RRRC6480
RRRC6490
RRRC6500
RRRC6510
RRRC6520
RRRC6530
RRRC6540
RRRC6550

```



```

C      REFCRE THE SWITCH WAS CONSIDERED
68C NISVT(MNEW) = NISV(MNEW)
   NISVT(MOLD) = NISV(MOLD)
   DC 685 J=1,NVARS
   SVCENT(MNEW,J) = SVCEN(MNEW,J)
685 SVCENT(MOLD,J) = SVCEN(MOLD,J)
69C CCNTINUE
691 CCNTINUE
C      FINISH WITH CLUSTER M: IF NO SWITCHES HAVE BEEN MADE, SET
C      JFLAG(M) = 2 AND GO TO NEXT CLUSTER: IF SWITCHES HAVE BEEN MADE,
C      ADJUST LSTSV AND SET JFLAG(M) = C AND GO TO NEXT CLUSTER
695 IF (JFLAG(M).EQ.0) GO TO 699
   JFLAG(M) = C
   GC TO 700
695 JFLAG(M) = 2
70C CCNTINUE
C      DCNE WITH ALL CLUSTERS: IFLAG = 1 MEANS SOME SWITCHES HAVE BEEN
C      MADE; GO BACK AND ITERATE
   IF (IFLAG.EC.1) GC TO 600
C      ALL DONE; ACCURATELY CALCULATE MEANS AND CRITERION AND CLPUT
   THE RESULTS
75C WRITE (6,90C)
90C FCRMAT (1,'THE FINAL CLUSTER SOLUTION IS ')
C      RECALCULATE CLUSTER CENTERS, WITHIN-CLUSTERS MATRIX, AND CRITERION
   VALUE
   DC 715 M=1,NGPS
   DC 715 J=1,NVARS
715 SVCEN(M,J) = 0.
   DC 720 I=1,NOBS
   M = IDATA(I)
   DC 720 J=1,NVARS
72C SVCEN(M,J) = SVCEN(M,J) + DATA(I,J)/NISV(M)
   CALL WCALC (SVCEN,NISV,NGPS,IDIR)
   IF (IFINE.EC.1) RETURN
   CALL CRITON (CRIT)
   IF (IFINE.EC.1) RETURN
   CALL THE OUTPUT ROUTINE
   CALL OUTPUT (CRIT)
   WRITE (6,901)
901 FCRMAT (1,0 END OF CLUSTER PROBLEM)
   RETURN
   END
C      SUBROUTINE OUTPUT (CRIT)
C      THIS SUBROUTINE PRINTS OUT THE CLUSTER SOLUTION
C      FOR EACH CLUSTER - THE CLUSTER SIZE

```

```

RRR06560
RRR06570
RRR06580
RRR06590
RRR06600
RRR06610
RRR06620
RRR06630
RRR06640
RRR06650
RRR06660
RRR06670
RRR06680
RRR06690
RRR06700
RRR06710
RRR06720
RRR06730
RRR06740
RRR06750
RRR06760
RRR06770
RRR06780
RRR06790

RRR06810
RRR06820
RRR06830
RRR06840
RRR06850
RRR06860
RRR06870
RRR06880
RRR06890
RRR06900
RRR06910
RRR06920
RRR06930
RRR06940
RRR06950
RRR06960
RRR06970
RRR06980
RRR06990
RRR07000
RRR07010
RRR07020
RRR07030

```



```

C      THE WITHIN CELLS MATRIX
C      THE CRITERION VALUE
C      THE CLUSTER CENTROID
C      THE OBSERVATIONS BELONGING TO THE CLUSTER

COMMON NOBS, NVARS, NGPS, ICRIT, NOSTAN, IDIST, IFINE, KTIME, ICENT(600),
1 DATA(600,20), T(20,20), B(20,20), W(20,20), WFCT(20,20), SVCEN(20,20),
2 ICATA(600), NISV(20), VMEAN(20), SC(20), XVEC(20), YVEC(20), ET(20,20),
3 NISVT(20), SVCENT(20,20), IDATAT(600), VEC(20,20), EIG(20)
UNSTANDARDIZE IF NECESSARY
IF (NOSTAN.EQ.0) GO TO 40
CC 20 M=1,NGPS
CC 20 J=1,NVARS
2C SVCEN(M,J) = SVCEN(M,J)*SD(J)
CC 30 J=1,NVARS
CC 30 K=J,NVARS
3C T(J,K) = SD(J)*T(J,K)*SC(K)
ICIR = 2
IF (ICIR.EQ.1) ICIR=1
CALL WCALC (SVCEN,NISV,NGPS,ICIR)
IF (IFINE.EQ.1) RETURN
CALL CRITON (CRIT)
IF (IFINE.EQ.1) RETURN
ALL THE OVERALL MEAN BACK INTO EACH OBSERVATION; UNSTANDARDIZE
THE DATA IF NECESSARY
4C CC 80 J=1,NVARS
CC 50 I=1,NCBS
IF (NOSTAN.EQ.1) DATA(I,J) = DATA(I,J)*SD(J)
5C DATA(I,J) = DATA(I,J) + VMEAN(J)
CC 60 M=1,NGPS
6C SVCEN(M,J) = SVCEN(M,J) + VMEAN(J)
CC CONTINUE
C WRITE OUT NISV, SVCEN
CC 115 M=1,NGPS
WRITE (6,900) M
WRITE (6,901) NISV(M)
WRITE (6,902) (SVCEN(M,J),J=1,NVARS)
WRITE (7,915) (SVCEN(M,J),J=1,NVARS)
WRITE (8F8.4, /8F8.4)
15 LSINC = NISV(M)
K = 0
CC 100 I=1,NOBS
IF (ICATA(I).NE.M) GO TO 10C
K = K+1
ICATAT(K) = IDENT(I)
CC CONTINUE
10C SCRT THE OBSERVATION IDENTIFICATIONS
C
RRR07C4C
RRR07050
RRR07C6C
RRR07C7C
RRR07C80
RRR07C90
RRR07100
RRR07110
RRR07120
RRR0713C
RRR07140
RRR0715C
RRR0716C
RRR07170
RRR0718C
RRR07190
RRR0720C
RRR07210
RRR0722C
RRR07230
RRR07240
RRR07250
RRR0726C
RRR0727C
RRR07280
RRR0729C
RRR07300
RRR0731C
RRR07320
RRR0733C
RRR07340
RRR0735C
RRR0736C
RRR0737C
RRR07380
RRR0739C
RRR0740C

RRR07410
RRR07420
RRR07430
RRR0744C
RRR07450
RRR0746C
RRR0747C
RRR07480
RRR0749C

```


RRR07500
RRR07510
RRR07520
RRR07530
RRR07540
RRR07550
RRR07560
RRR07570
RRR07580
RRR07590
RRR07600
RRR07610
RRR07620
RRR07630

RRR07640
RRR07650
RRR07660
RRR07670
RRR07680
RRR07690
RRR07700
RRR07710
RRR07720
RRR07730
RRR07740
RRR07750
RRR07760
RRR07770
RRR07780
RRR07790
RRR07800
RRR07810
RRR07820
RRR07830
RRR07840
RRR07850
RRR07860
RRR07870
RRR07880
RRR07890
RRR07900
RRR07910
RRR07920
RRR07930
RRR07940
RRR07950

```

101 L=L+1
102 IF (L.EC.NISV(M)) GO TO 110
    IF (L.L+1)
    IF (ICATAT(L).LE.ICATAT(LL)) GO TO 101
    LTEMP = IDATAT(L)
    ICATAT(L) = IDATAT(LL)
    ICATAT(LL) = LTEMP
    IF (L.EC.1) GO TO 102
    L=L-1
    GC TO 102
110 CCNTINUE THE IDENTIFICATIONS
    WRITE (6,904) (IDATAT(K),K=1,LSTAC)
    WRITE (7,920) (IDATAT(K),K=1,LSTNC)
920 FCRMAT(1615)
115 CCNTINUE THE WITHIN-CLUSTERS MATRIX
115 WRITE (6,905)
    CC 120 J=1,NVARS
    WRITE (6,906) (W(K,J),K=1,J)
120 CCNTINUE THE CRITERION CHOSEN BY THE USER
    GC TO (15C,160,17C,17C),ICRIT
15C WRITE OUT TRACE W
15C WRITE (6,907) CRIT
    RETURN
16C WRITE OUT DET W
    AND LOG (DET T / DET W)
16C CALL UPFCT (NVARS,T,M)
    DET = 1.
    CC 165 J=1,NVARS
    DET = DET*(J,J)
165 WRITE (6,912) CRIT
    CRIT = ALOG 10 ((DET**2)/(CRIT**2))
    WRITE (6,909) CRIT
    RETURN
17C FCF LARGEST FOOT AND PCTELLING'S TRACE CRITERIA, FINE EIGENVALUES
    FOR IB-GW=0.
    CC 175 J=1,NVARS
    CC 175 K=J,NVARS
    BT(J,K) = B(J,K)
175 BT(K,J) = BT(J,K)
    CRIT = 1.0 / CRIT
    CALL UTISUI (NVARS,BT,WFCF)
    CALL EIGN (NVARS,ET,EIG,VEC,IND)
    IF (IND.NE.C) GO TO 180
    WRITE OUT EIGENVALUES
    C

```



```

C      WRITE (6,91C) (EIG(J),J=1,NVARS)
C      IF (ICRIT.EC.4) GC TO 178
C      WRITE OUT LARGEST ROOT
C      WRITE (6,91E) CRIT
C      RETURN
C      WRITE OUT SUM OF EIGENVALUES
C      178 WRITE (6,914) CRIT
C      RETURN
C      ESCAPE ROUTE
C      180 IFINE = 1
C      WRITE (6,911) IND
C      RETURN
C      FCRMAT STATEMENTS
C      90C FCRMAT ('OCLUSTER',I2)
C      901 FCRMAT ('SIZE',I3)
C      902 FCRMAT ('CENTER',I5(F9.2,1X),/,18X,5(F9.2,1X))
C      903 FCRMAT ('THE OBSERVATIONS ARE',)
C      904 FCRMAT (8X,5I7,/,5X,5I7)
C      905 FCRMAT ('THE WITHIN GROUP MATRIX IS ')
C      906 FCRMAT ('ICE OF THE WITHIN-CLUSTERS MATRIX IS ',E12.6)
C      907 FCRMAT ('THE TRACE OF THE WITHIN-CLUSTERS MATRIX IS ',E12.6)
C      908 FCRMAT ('O DETERMINANT / DET W) = ',E12.6)
C      909 FCRMAT ('O LOG (DETERMINANT / DET W) ARE ',10F8.3)
C      910 FCRMAT ('O THE EIGENVALUES OF W(INVERSE) ARE ',I3)
C      911 FCRMAT ('O THE EIGENVALUES OF THE WITHIN-CLUSTERS MATRIX IS ',E12.6)
C      912 FCRMAT ('O THE DETERMINANT OF THE WITHIN-CLUSTERS MATRIX IS ',E12.6)
C      913 FCRMAT ('O THE LARGEST ROOT IS ',E12.6)
C      914 FCRMAT ('O THE TRACE OF W(INV)*B IS ',E12.6)
C      END
C      SUBROUTINE DISTCE (XVECS,YVECS,DIST)
C      THIS SUBROUTINE CALCULATES EUCLIDIAN, WEIGHTED EUCLIDIAN, CR
C      MAHALANOBIS DISTANCE (DEPENDING CN IDIST).
C      COMMON NOBS,NVARS,NGPS,ICRIT,NCSTAN,IDIST,IFINE,KTIME,ICENT(6C0),
C      1DATA(6C0,20),T(2C,20),R(20,20),W(20,20),WFCT(2C,20),SVCEN(20,2C),
C      2ICATA(600),NISV(20),VMEAN(20),SC(2C),XVEC(2C),YVEC(2C),E1(2C,2C),
C      3CJMSIGN,X(20),SVCENT(20,20),IDATAT(6C),VEC(20,20),EIG(20)
C      ICLR = IDIST+1
C      GC TO (10C,20C,30C),IDIR
C      WRITE (6,90C)
C      FCRMAT ('I ERROR IN DISTANCE CODE - COLUMN 10 OF PRCELEM CARD')
C      90C IFINE = 1
C      RETURN

```

```

RRRC756C
RRRC7570
RRRC758C
RRRC759C
RRRC8000
RRRC8010
RRRC8020
RRRC8030
RRRC8040
RRRC8C5C
RRRC8060
RRRC8C7C
RRRC8C80
RRRC8090
RRRC8100
RRRC8110
RRRC8120
RRRC8140
RRRC8170
RRRC818C
RRRC8190
RRRC820C
RRRC8210
RRRC8220
RRRC8230
RRRC824C
RRRC825C
RRRC8260
RRRC827C
RRRC8280
RRRC829C
RRRC8300
RRRC8310
RRRC8320
RRRC8330
RRRC8340
RRRC8350
RRRC8360
RRRC8370
RRRC838C
RRRC8390
RRRC840C
RRRC8410
RRRC8420
RRRC8430

```



```

C C
C C      CALCULATE EUCLIDIAN DISTANCE
100 DIST=0.
110 J=1,NVARS
120 DIST = DIST + (XVECS(J)-YVECS(J))**2
130 RETURN
C C
C C      CALCULATE EUCLIDIAN DISTANCE FOR STANDARDIZED VARIABLES
200 DIST=0.
210 J=1,NVARS
220 DIST = DIST + (XVECS(J)-YVECS(J))*(1.0/W(J,J))*(XVECS(J)-YVECS(J))
230 RETURN
C C
C C      CALCULATE MAHALANOBIS DISTANCE AS THE ELEMENTS OF
C C      L*(INV)(XVEC-YVEC) SQUARED WHERE L IS THE CHOLESKY FACTOR OF W
300 DIST=0.
310 J=1,NVARS
320 X(J) = XVEC(J) - YVECS(J)
330 CALL UTIRT (NVARS,1,WFACT,X)
340 J=1,NVARS
350 DIST = DIST + X(J)**2
360 RETURN
C C
C C      END
C C      SLEROUTINE CRITON (CRIT)
C C
C C      THIS SUBROUTINE CALCULATES THE CRITERIA VALUES
C C      CCMCN NOBS,NVARS,NGPS,ICRIT,NOSTAN,IDIST,IFINE,KTIME,IDENT(6CC),
101 IDATA(6CC,20),T(20,20),R(20,20),W(20,20),WFCT(20,20),SVCEN(20,20),
102 IICATA(6CC),NISV(20),VMEAN(20),SC(20),XVEC(20),YVEC(20),E1(20,20),
103 NISVT(20),SVCENT(20,20),IDATA(6CC),VEC(20,20),EIG(20)
900 GO TO (1CC,2CC,3CC,400),ICRIT
910 WRITE (6,90C)
920 FCMAT = 1
930 IFINE = 1
940 RETURN
C C
C C      CALCULATE TRACE W
100 CRIT = 0.
110 J=1,NVARS
120 CRIT = CRIT + W(J,J)
130 RETURN
C C
C C      CALCULATE DET W AS THE PRODUCT OF DIAGONAL ELEMENTS OF THE
C C      CHOLESKY FACTOR OF W

```



```

C      200    CET = 1., NVAR$
C      205    J=1, NVARS
C      205    DET = DET * WFCT(J,J)
C      205    CRIT = CET
C      205    RETURN
C
C      300    CALCULATE LARGEST ROOT AS L(INV)*B*L*(INV) WHERE L IS THE
C      300    CHCLESKY FACTOR OF W
C      300    CFIITERION IS RECIPROCAL OF LARGEST ROOT
C
C      300    CC 305    J=1, NVARS
C      300    CC 305    K=J, NVARS
C      300    BT(J,K) = B(J,K)
C      300    BT(K,J) = BT(J,K)
C      300    CALL UTISUI(NVAR$,BT,WFCT)
C      300    CALL EIGN(NVAR$,ET,EIG,VEC,IND)
C      300    CRIT = 1.C / EIG(1)
C      300    RETURN
C
C      400    CALCULATE HCTELLING'S TRACE AS SUM OF DIAGONAL ELEMENTS CF
C      400    L(INV)*B*L*(INV)
C      400    CRITERION IS RECIPROCAL OF THIS SUM
C
C      400    CC 405    J=1, NVARS
C      400    CC 405    K=J, NVARS
C      400    BT(J,K) = B(J,K)
C      400    BT(K,J) = BT(J,K)
C      400    CALL UTISUI(NVAR$,BT,WFCT)
C      400    CRIT = 0.
C      400    CC 410    J=1, NVARS
C      400    CRIT = CRIT + BT(J,J)
C      400    CRIT = 1.O / CRIT
C      400    RETURN
C      400    ENC
C      400    SLROUTINE WCALC(SV,NI,NGP,ICIR)
C
C      700    THIS SUBROUTINE CALCULATES THE WITHIN-CELLS MATRIX AND, IF
C      700    NECESSARY, THE CHCLESKY FACTOR CF THE WITHIN-CELLS MATRIX
C
C      800    COMMON NOBS,NVAR$,AGPS,ICRIT,NOSTA,N,IDIST,IFINE,KTIME,IDENT(6CC),
C      800    IDATA(6CC,20),T(20,20),B(20,20),W(20,20),WFCT(20,20),SVCEN(20,20),
C      800    IDATA(6CC),NISV(20),VMEAN(20),SC(20),XVEC(20),YVEC(20),BT(20,20),
C      800    NISVT(20),SVCENT(20,20),IDATAT(6CC),VEC(20,20),EIG(20)
C      800    DIMENSION SV(20,20),NI(20)
C
C      900    CALCULATE B AND THEN W = T - B
C      900    B = SUM NISV(M)*SVCEN(M)**2

```



```

C
100 C C 100 J=1,NVARS
100 C C 100 K=J,NVARS
100 C C 105 M=1,NGP
100 C C 105 J=1,NVARS
100 C C 105 K=J,NVARS
105 B T(J,K) = SV(M,J) * SV(M,K) * NI(M)
105 E(J,K) = E(J,K) + ET(J,K) * NI(M)
110 C C 110 J=1,NVARS
110 C C 110 K=J,NVARS
110 W(J,K) = T(J,K) - E(J,K)
C
C CALCULATE CHCLESKY FACTOR OF W IF ICIR = 2
C
115 C C TO (120,115),ICIR
115 C C 116 J=1,NVARS
115 C C 116 K=J,NVARS
116 W FCT(J,K) = W(J,K)
116 C C CALL UPFFCT (NVARS,WFCT,M)
120 I F (M.NE.0) GO TO 125
125 R ETURN
125 I F I N E=1
125 W R I T E (6,90C)
125 C C F C R M A T (1 THE WITHIN GROUPS MATRIX IS SINGULAR *)
125 C C R E T U R N
125 C C E N D
125 C C F U N C T I O N U R A N D (I R A N D)
125 C C
125 C C T H I S F U N C T I O N C A L C U L A T E S U N I F O R M L Y D I S T R I B U T E D R A N D O M N U M B E R S
125 C C B E T W E E N 0 A N D 1
125 C C 3 * 1 9 C O N G R U E N T I A L U N I F O R M R A N D O M N U M B E R G E N E R A T O R
125 C C
125 C C I R A N D = I R A N D * 1162261467
125 C C I F (I R A N D . G T . 0) G O T O 3
125 C C I R A N D = - I R A N D
125 C C U R A N D = F L O A T (I R A N D) * 0.4656612873E-9
125 C C R E T U R N
125 C C E N D
125 C C S U B R O U T I N E L P R F C T (N,A,M)
125 C C
125 C C R E P L A C E U P P E R T R I A N G L E O F A S Q U A R E P O S I T I V E D E F I N I T E M A T R I X A
125 C C B Y I T S C H O L E S K I F A C T O R
125 C C
125 C C D I M E N S I O N A (20,20)
125 C C C L E A R T H E E R R O R I N D I C A T O R
125 C C M = C
125 C C N I = N - 1

```

```

RRRC940C
RRRC941C
RRRC942C
RRRC943C
RRRC944C
RRRC945C
RRRC946C
RRRC947C
RRRC948C
RRRC949C
RRRC950C
RRRC951C
RRRC952C
RRRC953C
RRRC954C
RRRC955C
RRRC956C
RRRC957C
RRRC958C
RRRC959C
RRRC960C
RRRC961C
RRRC962C
RRRC963C
RRRC964C
RRRC965C
RRRC966C
RRRC967C
RRRC968C
RRRC969C
RRRC970C
RRRC971C
RRRC972C
RRRC973C
RRRC974C
RRRC975C
RRRC976C
RRRC977C
RRRC978C
RRRC979C
RRRC980C
RRRC981C
RRRC982C
RRRC983C
RRRC984C
RRRC985C
RRRC986C
RRRC987C

```



```

100 IF(N1) 23C,10C,10C
101 DC 220 K=1,N
102 AKK=A(K,K)
103 IF(K.EC.1) GO TC 12C
104 DC 110 J=2,K
105 AKK=AKK-A(J-1,K)**2
106 IF(A(K,K)) 14C,14C,130
107 IN THE CASE OF A CCVARIANCE MATRIX AKK/A(K,K) IS 1-F**2 WHERE
108 R IS MULT CORRELATION OF VARIABLE K WITH ALL PRECEDING VARIABLES.
109 IF(AKK/A(K,K)).GE..001) GO TC 150
110 N=K
111 AKK=0
112 AKK=SCRT(AKK)
113 A(K,K)=AKK
114 IF(K.EC.N) GO TC 23C
115
116 DC 220 I=K,N1
117 AKI=A(K,I+1)
118 IF(K.EC.1) GO TC 190
119 DC 180 J=2,K
120 AKI=AKI-A(J-1,K)*A(J-1,I+1)
121 IF(AKK) 210,200,210
122 A(K,I+1)=0.0
123 GC TO 220
124 A(K,I+1)=AKI/AKK
125 CC CONTINUE
126 RETURN
127 END
128 SUBROUTINE LTIRT(N,N,S,B)
129
130 INVERSE CF UPPER (S) TRANSPOSED TIMES RECTANGLE B TRANSFCSED.
131
132 DIMENSION S(20,20),B(1,20)
133 DC 130 J=1,N
134 DC 130 I=1,N
135 SUM=0.0
136 IF(S(I,I)) 90,120,90
137 SUM=B(J,I)
138 IM1=I-1
139 IF(IM1) 120,120,100
140 DC 110 K=1,IM1
141 SUM=SUM-S(K,I)*B(J,K)
142 SUM=SUM/S(I,I)
143 B(J,I)=SUM
144 RETURN
145 END
146 SUBROUTINE LTISUI (N,A,B)

```



```

C C      UPPER (B) TRANSPCSE INVERSE TIMES A TIMES UPPER (B) INVERSE.
C C      DIMENSION A(20,20),B(20,20)
C C      NCITE THAT PCSTMULT IS CARRIED OUT ON FINAL VALUES LEFT BY PREMULT
C C      DC 200 I=1,N
C C      I=I-1
C C      NCITE THAT PREMULT ON RIGHT HALF CF ROW IS SAME AS PCSTMULT
C C      CN LOWER HALF OF COLUMN - EXCEPT FCR DIAG TERM WHICH IS THE
C C      FINAL VALUE LEFT BY PREMULT BEFORE POSTMULT
C C      CC 130 J=1,N
C C      IF(I1) 120,12C,10C
C C      CC 11C K=1,I1
C C      A(J,I) = A(J,I) - B(K,I)*A(J,K)
C C      IF A(J,I) = A(J,I)/B(I,I)
C C      IF(B(I,I) .EQ. 0.0) A(J,I) = 0.0
C C      NCITE THAT ELEMENTS IN LEFT HALF CF ROW ARE FINAL FCR PREMULT
C C      NCITE THAT DIAG ELEMENT WAS PREVIOUSLY THE FINAL RESULT CF
C C      FREMULT. NCW WE MAKE THE FINAL RESULT OF PCSTMULT.
C C      CC 200 J=1,I
C C      J1=J-1
C C      IF(J1) 16C,16C,14C
C C      HORIZONTAL BRANCH CF INNER PRODUCT EXCLUDING DIAG TERM
C C      CC 15C K=1,J1
C C      A(I,J) = A(I,J) - B(K,I)*A(J,K)
C C      IF(I1-J) 19C,170,170
C C      VERTICAL BRANCH CF INNER PRODUCT INCLUDING DIAG TERM
C C      CC 180 K=J,I1
C C      A(I,J) = A(I,J) - B(K,I)*A(K,J)
C C      A(I,J) = A(I,J)/B(I,I)
C C      IF(B(I,I) .EQ. 0.0) A(I,J) = 0.0
C C      RETURN
C C      SUBROUTINE EIGN(N,A,EIG,VEC,INC)
C C      NN= SIZE OF MATRIX
C C      A= MATRIX (ONLY LOWER TRIANGLE IS USED + THIS IS DESTROYED)
C C      EIG = RETURNED EIGENVALUES IN ALGEBRAIC DESCENDING ORDER
C C      VEC = RETURNED EIGENVECTORS IN COLUMNS
C C      INC = ERROR RETURN INDICATOR
C C      0 FOR NORMAL RETURN
C C      1 SUM CF EIGENVALUES NOT EQUAL TO TRACE
C C      2 SUM CF EIGENVALUES SQUARED NOT EQUAL TO NORM
C C      3 BCTH CF THESE ERRORS
C C      DIMENSION A(20,20),GAMMA(20),BETA(20),BETASC(20),EIG(20)
C C      THE FOLLOWING DIMENSIONED VARIABLES ARE EQUIVALENCED
C C      DIMENSION P(19),Q(19)
C C      EQUIVALENCE (P(1),BETA(1)),(Q(1),EETA(1))
C C      DIMENSION IFCV(20),IVPOS(20),ICRC(20)
RRR1C36C
RRR1C37C
RRR1C38C
RRR1C39C
RRR1C40C
RRR1C41C
RRR1C42C
RRR1C43C
RRR1C44C
RRR1C45C
RRR1C46C
RRR1C47C
RRR1C48C
RRR1C49C
RRR1C50C
RRR1C51C
RRR1C52C
RRR1C53C
RRR1C54C
RRR1C55C
RRR1C56C
RRR1C57C
RRR1C58C
RRR1C59C
RRR1C60C
RRR1C61C
RRR1C62C
RRR1C63C
RRR1C64C
RRR1C65C
RRR1C66C
RRR1C67C
RRR1C68C
RRR1C69C
RRR1C70C
RRR1C71C
RRR1C72C
RRR1C73C
RRR1C74C
RRR1C75C
RRR1C76C
RRR1C77C
RRR1C78C
RRR1C79C
RRR1C80C
RRR1C81C
RRR1C82C
RRR1C83C

```



```

      EQLVLENCE (IPOSV(1),GAMMA(1)),(IVPOS(1),BETA(1)),
      1(IICRD(1),BETASQ(1))
      N=NN
      C RESET ERRCR RETURN INDICATOR
      INC=0
      IF(N.EC. 0) GO TC 560
      NJ=N-1
      N2=N-2
      C COMPUTE THE TRACE AND EUCLIDIAN NCRM OF THE INPUT MATRIX
      C LATER CHECK AGAINST SUM AND SUM CF SQUARES CF EIGENVALUES
      ENCRM=0.
      TRACE=0.
      DO 100 J=1,N
      DO 100 I=J,N
      ENCRM=ENCRM+A(I,J)**2
      TRACE=TRACE+A(J,J)
      100 ENCRM=ENCRM-.5*A(J,J)**2
      110 ENCRM=ENCRM+ENORM
      GAMMA(1)=A(1,1)
      IF(N2) 280,270,120
      120 DO 260 NR=1,N2
      E=A(NR+1,NR)
      S=C.
      DO 130 I=NR,N2
      S=S+A(I+2,NR)**2
      130 PREPARE FOR POSSIBLE BYPASS OF TRANSFORMATION
      A(NR+1,NR)=C.
      IF(S) 250,250,140
      S=S+B*B
      140 SGN=+1.
      IF(B) 150,160,160
      SGN=-1.
      SCRTS=SCRT(S)
      C=SGN/(SCRTS+SCRTS)
      TEMP=SCRT(.5+B*C)
      A(NR)=TEMP
      A(NR+1,NR)=TEMP
      C=C/TEMP
      B=-SGN*SCRTS
      C IS FACTOR CF PROPORTIONALITY. NCM COMPUTE AND SAVE VECTOR.
      EXTRA SINGLY SUBSCRIPTED VECTOR USED FOR SPEED.
      170 I=NR,N2
      TEMP=D*A(I+2,NR)
      W(I+1,NR)=TEMP
      W(I+2,NR)=TEMP
      C PREMULTIPLY VECTOR W BY MATRIX A TO OBTAIN F VECTOR.
      C SIMULTANEOUSLY ACCUMULATE DOT PRODUCT WP,(THE SCALAR K)
      WTAW=0.

```



```

      CC 220 I=NR,N1
      SUM=0.
      CC 180 J=NR,I
      SUM=SUM+A(I+1,J+1)*W(J)
      JJ=I+1
      IF(N1-I) 210,150,190
      CC 200 J=I1,N1
      SUM=SUM+A(J+1,I+1)*W(J)
      P(I)=SUM
      W1A=W1A+W1A*W(I)
      F VECTOR AND SCALAR K NOW STORED. NEXT COMPLETE Q VECTOR
      CC 230 I=NR,N1
      Q(I)=P(I)-W1A*W(I)
      C
      C NEW FORM PAF MATRIX, REQUIRED PART
      CC 240 J=NR,N1
      QJ=Q(J)
      WJ=W(J)
      CC 240 I=J,N1
      A(I+1,J+1)=A(I+1,J+1)-2.*(W(I)*QJ+WJ*Q(I))
      240 BETA(NR)=B
      250 BETASQ(NR)=E*B
      260 GAMMA(NR+1)=A(NR+1,NR+1)
      270 E=A(N,N-1)
      BETA(N-1)=B
      BETASQ(N-1)=B*B
      GAMMA(N)=A(N,N)
      280 BETASQ(N)=0
      C ACJGIN AN IDENTITY MATRIX TC BE POSTMULTIPLIED BY ROTATIONS.
      CC 300 I=1,N
      CC 290 J=1,N
      290 VEC(I,J)=0.
      300 VEC(I,I)=1.
      SUM=0.
      NFA=1
      CC TO 400
      SUM=SUM+SHIFT
      310 CCSA=1.
      GF=GAMMA(1)-SHIFT
      PF=G
      FEES=PP*PP+FEETASQ(1)
      PFER=SQRT(PFES)
      CC 370 J=1,N
      CCSAP=CCSA
      IF(PFBS .NE. 0.) GC TC 320
      SINAA=0.
      SINAA2=0.
      CCSA=1.

```

```

RRR111320
RRR111330
RRR111340
RRR111350
RRR111360
RRR111370
RRR111380
RRR111390
RRR111400
RRR111410
RRR111420
RRR111430
RRR111440
RRR111450
RRR111460
RRR111470
RRR111480
RRR111490
RRR111500
RRR111510
RRR111520
RRR111530
RRR111540
RRR111550
RRR111560
RRR111570
RRR111580
RRR111590
RRR111600
RRR111610
RRR111620
RRR111630
RRR111640
RRR111650
RRR111660
RRR111670
RRR111680
RRR111690
RRR111700
RRR111710
RRR111720
RRR111730
RRR111740
RRR111750
RRR111760
RRR111770
RRR111780
RRR111790

```



```

42C SHIFT=R1
43C GO TO 310
C EIG(I)=GAMMA(I)+SLM
INITIALIZE AUXILIARY TABLES REQUIRED FOR REARRANGING THE VECTORS
DC 440 J=1,N
IVPCSV(J)=J
44C ICRD(J)=J
C USE A TRANSPOSITION SCRT TO ORDER THE EIGENVALUES
M=N
GC TO 470
CC 460 J=1,N
IF(EIG(J).EQ.EIG(J+1)) GO TO 46C
TEMP=EIG(J)
EIG(J)=EIG(J+1)
EIG(J+1)=TEMP
ITEMP=ICRD(J)
ICRD(J)=ICRD(J+1)
ICRD(J+1)=ITEMP
CC CONTINUE
46C M=M-1
47C IF(M.EQ.0) GO TO 45C
IF(N1.EQ.C) GO TO 500
CC 490 L=1,N1
NV=IPCSV(NV)
NF=IPCSV(NF)
IF(NP.EQ.L) GO TO 490
LV=IVPOS(L)
IVPCS(NP)=LV
IPOSV(LV)=NF
CC 480 I=1,N
TEMP=VEC(I,L)=VEC(I,NP)
VEC(I,L)=TEMP
VEC(I,NP)=TEMP
CC CONTINUE
48C ESSL=0.
49C BACK TRANSFORM THE VECTORS OF THE TRIPLE DIAGONAL MATRIX
50C K=N1
C K=K-1
51C IF(K=1) LE. 0) GO TO 540
SLM=0.
CC 520 I=K,N1
SLM=SUM+VEC(I+1,NRR)*A(I+1,K)
52C SLM=SUM+SUM
DC 530 I=K,N1
53C VEC(I+1,NRR)=VEC(I+1,NRR)-SUM*A(I+1,K)

```



```

54C CC TO 51C
55C ESUM=ESUM+EIG(NRR)*2
    ESQ=ESQ+EIG(NRR)*2
    TEMP=ABS(512*TRACE)+TEMP)-TEMP .NE. 0.) INC=INC+1
    IF((ABS(TRACE-ESUM)+TEMP)-TEMP .NE. 0.) INC=INC+2
    IF((ABS(ENORM-ESSC)+TEMP)-TEMP .NE. 0.) INC=INC+2
56C RETURN
    END

```

```

RRR1276C
RRR1277C
RRR1278C
RRR1279C
RRR1280C
RRR1281C
RRR1282C
RRR1283C
RRR1284C

```



```

Appendix C:
RUN NAME
VARIABLE LIST
INPLT MEDIUM
INPLT FORMAT
N OF CASES
VAR LABELS

FACTCR
CPTICNS
STATISTICS
REAC INFLT DATA
DATA
.
.
.
.
FINISH

PRINCIPAL CCMPONENT ANALYSIS OF TANKS
NATIONS,X1 TC X10
CARD
FIXED(1X,F2.0,5X,8F8.3/8X,2F8.3)
24
NATIONS TYPE OF TANK/
X1 WEIGHT/X2 LENGTH/X3 WIDTH/X4 HEIGHT/X5 ROAD SPEED/
X6 TRENCH CRCSING/X7 GRCLNC PRESSURE/X8 MAX ARMAMENT/
X9 GROUND CLEARANCE/X10 FCWER TO ENGINE RATIC/
VARIABLES=X1 TO X10/TYPE=PA1/NFACTCRS=10/
8,11
ALL

```


Appendix D:

```

RUN NAME LIST
VARIABLES
N CF CASES
INPLT MEDIUM
RECCCE
VAR LABELS

DISCRIMINANT
CPTICNS
STATISTICS
REAC INFLT DATA
DATA
.
.
.
.
.
FINISH

TANK DISCRIMINANT
NATIONS,X1 TC X10
24
FIXED(1X,F2,C,5X,8F8.3/6X,2F8.3)
NATIONS(11,THRU 13=1)(15,16,17,24,25,26,30,31,32,33=2)
(19,20,21,22,23,28,29,=3)(14,18,27,34=4)
NATIONS TYPE OF TANK/
X1 WEIGHT/X2 LENGTH/X3 WIDTH/X4 HEIGHT/X5 ROAD SPEED/
X6 TRENCHE CROSSING/X7 GROUND PRESSURE/X8 MAX ARMAMENT/
X9 GROUND CLEARANCE/X10 FCW/ER TO ENGINE RATIC/
GRUPS=NATIONS(1,4)/VARIABLES=X1 TC X10/
ANALYSIS=X1 TO X10/METHOD=MAHAL/
5,6,7,9,11,12
ALL

```


BIBLIOGRAPHY

1. Aiken, J. W., "Development of Cluster Analysis for Student Opinion Data", Master's Thesis, Naval Post-graduate School, Monterey, CA, 1979.
2. Anderson, T. W., An Introduction to Multivariate Statistical Analysis, Wiley, 1958.
3. Anderberg, M. R., Cluster Analysis for Applications, Academic Press, 1973.
4. Barr, D. R., and Richards, F. R., Utility Assessment Methodology (Report on Relative Utility Score of the AN-TPQ/27). System Exploration Inc., Monterey, CA, 1980.
5. Eisenheis, R. A., and Avery, R. B., Discriminant Analysis and Classification Procedures, Lexington Books, 1972.
6. Friedman, H. P., and Rubin, J., "On Some Invariant Criteria for Grouping Data", *Journal of the American Statistical Association*, Vol. 62: 320, Dec. 1967.
7. Giri, N. C., Multivariate Statistical Inference, Academic Press, 1977.
8. Gnanadesikan, R., Methods for Statistical Data Analysis of Multivariate Observations, Wiley, 1977.
9. Green, P. E., and Carroll, J. D., Mathematical Tools for Applied Multivariate Analysis, Academic Press, 1976.
10. Hartigan, J. A., Clustering Algorithms, Wiley, 1975.
11. Kandell, M. G., The Advanced Theory of Statistics, Vol. III, Charles Griffin and Company, 1947.
12. Keeney, R. L., and Raiffa, H., Decisions with Multiple Objectives, Wiley, 1976.
13. Klecka, W. R., "Discriminant Analysis", SPSS (Statistical Package for the Social Sciences), McGraw Hill, 1975.
14. Kim, J. O., "Factor Analysis", SPSS (Statistical Package for the Social Sciences), McGraw Hill, 1975.

15. MacQueen, J., "Some Methods for Classification and Analysis of Multivariate Observations", Paper presented at Fifth Berkeley Symposium on Mathematical Statistics and Probability, University of California, 1965-1966.
16. McRae, D. J., Clustering Multivariate Observations, Doctoral Dissertation, University of North Carolina, Chapel Hill, N.C., 1973.
17. Morrison, D. F., Multivariate Analysis: Technique for Educational and Psychological Research, Wiley, 1971.
18. The IMSL Library, Vol. 3, IMSL, Inc., 1979.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center (DTIC) Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3. Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93940	4
4. Professor F. R. Richards, Code 55Rh (Thesis Advisor) Department of Operations Research Naval Postgraduate School Monterey, California 93940	2
5. Professor D. R. Barr, Code 55Bn (Second Reader) Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
6. LTC Jin Ki Lee, ROK Army (Student) 6-601, Gun-In Apt, Dong-Bing-Go Dong Yong-San, Seoul Korea	5

Thesis
L39
c.1

Lee

Classification tech-
niques for multivari-
ate data analysis.

187912

tech-
iate

22 OCT 81
16 FEB 82
25 APR 87
10 APR 89

27252
27511
31585
32697

585
897

Thesis
L39
c.1

Lee

Classification tech-
niques for multivari-
ate data analysis.

187912

thesL39
Classification techniques for multivariate



3 2768 001 03163 6
DUDLEY KNOX LIBRARY